

# **Interaktive Lageregelung bei Nacht mit dem Mikrosatelliten DLR-TUBSAT**

vorgelegt von  
Dipl.-Ing. Jörn-Hendrik Bleif  
aus Berlin

An der Fakultät V – Verkehrs- und Maschinensysteme –  
der Technischen Universität Berlin  
zur Erlangung des akademischen Grades  
Doktor der Ingenieurwissenschaften (Dr.-Ing.)  
vorgelegte Dissertation

Promotionsausschuss:

Vorsitzender: Prof. Dr.-Ing. H. Linde  
Gutachter: Prof. Dr.-Ing. U. Renner  
Gutachter: Prof. em. Dr.-Ing. R. Prieb

Tag der wissenschaftlichen Aussprache: 24.05.2004



# Danksagung

Ich möchte mich an dieser Stelle bei allen direkt oder indirekt an der Entstehung dieser Dissertation beteiligten Personen bedanken.

Zunächst geht mein Dank an Prof. Renner, der mir die Promotion auf einem sehr spannenden Forschungsgebiet ermöglicht hat und mir den DLR-TUBSAT für meine Arbeit uneingeschränkt zur Verfügung gestellt hat.

Weiterhin möchte ich mich bei Prof. Prieps für die Übernahme der Funktion als zweiter Gutachter bedanken. Ebenso geht mein Dank an Prof. Linde für die Übernahme des Vorsitzes des Promotionsausschusses.

Den Mitarbeitern des DFD-Neustrelitz, insbesondere den Herren Richter und Dr. Schwarz, möchte ich für die gute Zusammenarbeit und die Bereitstellung der Antennenanlagen des DFD zum Empfang der DLR-TUBSAT-Bilddaten danken.

Auch bei Herrn Dudek möchte ich mich für die freundliche Unterstützung und die Nutzung seiner Antennenanlage bedanken.

Zuletzt geht mein Dank noch an die Herren Nel, Strümpfer und Prof. Du Plessis von der Universität Stellenbosch und an die Herren van Staden und Korki vom OTB in Overberg für die gute Zusammenarbeit während meines Forschungsaufenthalts in Südafrika.

## Abstract

Primäres Ziel dieser Forschungsarbeit war es, den Satelliten DLR-TUBSAT für die interaktive Lageregelung bei Nacht einsetzbar zu machen und die Funktionsfähigkeit des Systems anhand von Lageregelungsexperimenten im Orbit zu demonstrieren. Unter interaktiver Lagerregelung versteht man das 1999 an der TU-Berlin erstmals vorgestellte Lagerregelungsverfahren, bei dem ein Mensch in der Bodenstation mittels eines Joysticks Steuerbefehle an den Satelliten sendet, während er vor sich auf einem Bildschirm das vom Satelliten übertragene Live-Videobild sehen kann. Er bekommt dadurch in Echtzeit die Reaktion des Satelliten auf die Steuerbefehle vermittelt.

Hinter diesem Ziel steckte die Motivation, das zeitliche Auflösungsvermögen des Satelliten zu erhöhen, sowie neue Anwendungsmöglichkeiten zu erschließen.

Neben diesem Hauptziel sollte zusätzlich das Potential der DLR-TUBSAT-Nutzlastoptiken für Nachtaufnahmen untersucht werden.

Das Kernproblem, welches zur Erreichung dieser beiden Ziele gelöst werden musste, war das der Vorstabilisierung des Satelliten bei Nacht. Unter Vorstabilisierung versteht man das auf einige Grad genaue Ausrichten des Satelliten auf ein Zielgebiet, welches anschließend interaktiv erkundet werden soll.

Um dieses Problem zu lösen, wurde eine komplexe Vorstabilisierungssoftware entwickelt welche sich aus boden- und bordseitigen Komponenten zusammensetzt. Mit der Software lässt sich der Satellit in drei Schritten von einer beliebigen unbekannten Lage zu Beginn des Überflugs, der sogenannten Lost-In-Space-Konfiguration, in den geforderten vorstabilisierten Zustand überführen. Die drei Schritte sind:

- Aufnahme und Übertragung eines Sternbilds zur Bodenstation
- Ermittlung der Lage des Satelliten durch Auswertung des Sternbilds
- Berechnung und Kommandierung der notwendigen Schwenkmanöver um von der ermittelten zur gewünschten vorstabilisierten Lage zu gelangen

Zu jedem dieser Schritte wurden zunächst eigenständige Programme entwickelt, welche anschließend zu der integrierten Vorstabilisierungssoftware zusammengefügt wurden.

Die für diese Arbeit entwickelte Software zur Berechnung und Kommandierung der Manöver wurde als Echtzeitsimulator unter Einbindung von OpenGL-Grafikroutinen realisiert. Bei der Kommandierung der Manöver werden diese von der Simulation mitgekoppelt, sodass der Operator den gesamten Vorstabilisierungsvorgang aus der simulierten Satellitenperspektive mitverfolgen kann, was ihm die Interpretation der Szene im echten Videobild erleichtert.

Neben der Lösung des Problems der Vorstabilisierung bei Nacht wurde im Rahmen dieser Dissertation auch die interaktive Steuerung selbst optimiert. Hierfür wurde wiederum eine neue Software entwickelt. Diese wurde vor dem Einsatz bei Nacht erfolgreich am Tage in Südafrika getestet. Es konnte demonstriert werden, dass es diese Software ermöglicht, mit dem 1000mm-Objektiv über mehrere Minuten einer Küstenlinie zu folgen, ohne dieselbe aus dem Blickfeld zu verlieren.

Die Genauigkeit der integrierten Vorstabilisierungssoftware wurde zunächst durch Manöver von einem Sternbild zu einem anderen getestet. Für die sich anschließenden interaktiven Lageregelungstests wurde Berlin als Ziel ausgewählt. Die Funktion des Vorstabilisierungssystems konnte durch das Erreichen von Berlin mit dem 50mm-Objektiv bei mehreren aufeinanderfolgenden Überflügen nachgewiesen werden. Auch die Funktion der interaktiven Steuerung konnte durch das erfolgreiche zentrieren des Ziels im Bild nachgewiesen werden.

# Inhaltsverzeichnis

1	EINLEITUNG	1-1
2	STAND DER TECHNIK	2-4
2.1	Interaktive Lageregelung bei Tag mit DLR-TUBSAT	2-4
2.2	Bestehende Systeme für die Erdbeobachtung bei Nacht	2-6
3	ENTWICKLUNG DES STERNSENSORS	3-10
3.1	Ausgangssituation	3-10
3.1.1	Kamera Hardware	3-10
3.1.2	Ursprüngliche Sternerkennung	3-12
3.2	Sternerkennung trotz Strahlenschäden	3-13
3.2.1	Aufnahme der Sternbilder	3-13
3.2.2	Flächenkriterium und Amplitudenkriterium	3-13
3.2.3	Bordseitige Datenreduktion	3-17
3.2.4	Bodenseitige Datenvisualisierung	3-21
4	SOFTWARE ZUR STERNMUSTERERKENNUNG	4-24
4.1	Sternmustererkennung beim Sternsensor KM 1301	4-24
4.2	Halbautomatische Sternmustererkennung	4-26
4.2.1	Koordinatensysteme	4-26
4.2.2	Sternauswahl im reduzierten Sternbild	4-29
4.2.3	Verwendete Datenstrukturen	4-29
4.2.4	Erstellung des Musterkatalogs	4-31
4.2.5	Mustererkennung	4-33
4.2.6	Bestimmung der Satellitenlage	4-40
5	SIMULATION UND MANÖVER	5-44
5.1	Orbit Simulator	5-44
5.1.1	Allgemeines	5-44
5.1.2	Modellierung mit Open GL	5-44
5.1.3	Lagebeschreibung	5-51
5.2	Manöverberechnung	5-53
5.2.1	Berechnung der Schwenkwinkel	5-53
5.2.2	Benutzerschnittstelle	5-57
6	INTEGRIERTE VORSTABILISIERUNGSSOFTWARE	6-59
6.1	Benutzerinterface	6-59
6.2	Ablauf der Vorstabilisierung	6-60

---

7	INTERAKTIVE STEUERUNGSSOFTWARE	7-68
7.1	Funktionsprinzip	7-68
7.2	Orbiterfahrung	7-70
8	FLUGERGEBNISSE	8-74
8.1	Vorversuche	8-74
8.1.1	Driftmessung	8-74
8.1.2	Schwenkmanöver von Sternbild zu Sternbild	8-76
8.2	Interaktive Lageregelung bei Nacht	8-78
8.2.1	Vorbemerkung zum Bodensegment	8-78
8.2.2	Vorstabilisierung	8-78
8.2.3	Interaktive Lageregelung mit der 50mm Optik	8-80
8.2.4	Interaktive Lageregelung mit der 1000mm Optik	8-82
8.3	Potential der Objektive für Nachtaufnahmen	8-86
8.3.1	16mm-Objektiv	8-86
8.3.2	50mm-Objektiv	8-87
8.3.3	1000mm-Objektiv	8-88
9	ZUSAMMENFASSUNG UND AUSBLICK	9-89
9.1	Zusammenfassung	9-89
9.2	Ausblick	9-91
ANHANG		
	Symbolverzeichnis	A-93
	Literaturverzeichnis	A-95

# 1 Einleitung

Das Ziel dieser Dissertation ist die Erweiterung des Systems DLR-TUBSAT für den Einsatz bei Nacht. Der Satellit ist im Rahmen einer Kooperation zwischen der Technischen Universität Berlin und dem Deutschen Zentrum für Luft- und Raumfahrt entwickelt worden. Der Start erfolgte im Mai 1999.

Mit diesem System wurde erstmals eine neue Form der Lageregelung für Mikrosatelliten demonstriert, die sogenannte interaktive Lageregelung. Hierbei entscheidet ein



**Abbildung 1-1:** Mikrosatellit DLR-TUBSAT

Mensch am Boden (im Folgenden als Operator bezeichnet) anhand eines Live-Videobildes, das vom Satelliten zum Boden übertragen wird, welche Schwenkmanöver ausgeführt werden sollen. Gesteuert werden die Manöver schließlich mit Hilfe eines Joysticks. Mit diesen Joystickkommandos ist der Operator in der Lage, nach einem Ziel zu suchen, dessen Position unbekannt ist, z.B. dem Absturzort eines Flugzeuges. Um die Joysticksteuerung einzuleiten, wird zunächst die schnelle Bewegung

des Satelliten über Grund, von etwa 7 km/s, durch die Kommandierung einer langsamen Gegenrotation um die Nickachse des Satelliten kompensiert. Beim Betrachter des Videobildes entsteht dadurch der Eindruck, der Satellit würde wie ein Hub-schrauber auf der Stelle stehen. Aus dieser Situation kann der Operator ein Gebiet dann mit dosierten Joystickkommandos über einen Zeitraum von einigen Minuten nach dem Ziel absuchen, oder ein gefundenes Ziel im Blickfeld halten. Die Forschungen zum Thema interaktive Lageregelung wurden an der TU-Berlin bis Ende 2000 von Dr.-Ing. Stefan Schulz durchgeführt, an dessen Dissertation [1] diese Forschungsarbeit anschließt.

Im Rahmen dieser Arbeit soll ein Verfahren entwickelt werden, welches die Anwendung der interaktiven Lageregelung auch bei Nacht ermöglicht. Eine Nutzung der Nachtüberflüge des Satelliten würde zu einer Verbesserung des zeitlichen Auflösungsvermögens des Satelliten führen. So könnten bspw. Waldbrände bei Tag und bei Nacht aufgenommen werden und somit ihre Entwicklung genauer verfolgt werden.

Eine weitere potentielle Einsatzmöglichkeit der interaktiven Lageregelung bei Nacht stellt die Verkehrsüberwachung dar. Die theoretische Auflösung des 1000mm-Objektivs an Bord von DLR-TUBSAT von 6 m reicht zwar nicht aus, um einzelne Fahrzeuge zu erkennen, es ist aber denkbar, dass bei Nacht die Scheinwerferkegel der Fahrzeuge aufgelöst werden können. In diesem Fall könnte man die Tatsache nutzen, dass der Satellit mit der Kamera über mehrere Minuten auf dem selben Zielgebiet verweilen kann, um z.B. ruhenden von fließendem Verkehr auf Autobahnen zu unterscheiden.

Um die interaktive Lageregelung bei Nacht zu ermöglichen, muss vor allem das Problem der Vorstabilisierung des Satelliten gelöst werden. Unter Vorstabilisierung versteht man das grobe, auf einige Grad genaue, Ausrichten des Satelliten auf ein Zielgebiet, ausgehend von einer beliebigen unbekannten Lage, der sogenannten Lost-In-Space-Konfiguration. Hierzu müssen Hard- und Software des Bodensegments, sowie die Software mehrerer Komponenten des Satelliten entsprechend modifiziert werden.

Neben der interaktiven Lageregelung soll im Rahmen dieser Arbeit auch das Potential der DLR-TUBSAT-Optiken für die Erdbeobachtung bei Nacht erforscht werden. Von besonderem Interesse ist hierbei das 1000mm-Objektiv, da die bestehenden Satellitensysteme welche für Nachtaufnahmen eingesetzt werden können, ausschließlich mit niedriger (bspw. DMSP mit 500 m) oder mittlerer Auflösung (bspw. LANDSAT mit 30 m) arbeiten. Mit Nachtaufnahmen sind in diesem Zusammenhang Aufnahmen im sichtbaren Spektralbereich gemeint.

Das DMSP System ist bereits seit den 60er Jahren in Betrieb und dient neben seiner militärischen Verwendung bis heute auch wissenschaftlichen Anwendungen. Bei-



spiele hierfür sind die Bestimmung der Bevölkerungsverteilung in Ballungsräumen [2] oder die Erforschung der sogenannten Lichtverschmutzung des Nachthimmels durch Streulicht auf der Basis von Satellitenbilddaten [3]. Hochauflösende Nachtbilder könnten in Zukunft dazu beitragen, diese Forschungen und die daraus gewonnenen Erkenntnisse zu verfeinern.

Wie schon erwähnt knüpft diese Arbeit inhaltlich an andere an der TU-Berlin veröffentlichte wissenschaftliche Arbeiten an. Es wird daher auf eine detaillierte Beschreibung des Mikrosatelliten DLR-TUBSAT verzichtet. Ausgenommen hiervon ist das Nutzlastkamarasystem des Satelliten, das eine zentrale Rolle bei dem implementierten Verfahren zur Vorstabilisierung bei Nacht spielt. Für weitergehende technische Informationen zu den Subsystemen wird an dieser Stelle auf die anderen wissenschaftlichen Arbeiten zum Thema DLR-TUBSAT [1][4] verwiesen.

Die Dissertation ist folgendermaßen gegliedert: Zunächst wird in Kapitel 2 der Stand der Technik aufgezeigt. Einerseits soll ein Überblick über die interaktive Lageregelung bei Tag [1] vermittelt werden. Hierbei wird auch begründet, warum die verwendete Lageregelung nicht auf Nachtüberflüge anwendbar ist. Andererseits sollen bestehende Satellitensysteme für die Erdbeobachtung bei Nacht näher vorgestellt werden.

Die Vorstabilisierung des Satelliten auf das Gebiet, welches in der Folge interaktiv mit dem Joystick erkundet werden soll, stellt den Kern dieser Forschungsarbeit dar. Ihr sind daher gleich mehrere Kapitel gewidmet. Die Kapitel 3,4 und 5 enthalten ausführliche Beschreibungen der für die Vorstabilisierung wichtigen boden- und bordseitigen Softwarekomponenten. Kapitel 6 beschreibt die fertige Manöversoftware, welche durch Integration der einzelnen Komponenten entstanden ist.

Kapitel 7 beschäftigt sich mit der Software zur interaktiven Lageregelung und deren Unterschieden zu der Software welche von Schulz [1] entwickelt wurde.

Kapitel 8 beschreibt schließlich die durchgeführten Lageregelungsexperimente, welche sich in die Kategorien Vorversuche und interaktive Lageregelung aufteilen. Außerdem fasst dieses Kapitel noch die durch die Experimente gewonnenen Erfahrungen bezüglich des Auflösungsvermögens der verschiedenen Nutzlastobjekte an Bord von DLR-TUBSAT bei Nacht zusammen.

## 2 Stand der Technik

### 2.1 Interaktive Lageregelung bei Tag mit DLR-TUBSAT

Die von Schulz [1] entwickelte Lageregelungsstrategie benutzt für die Akquisition oder Vorstabilisierung die Solarzellen des Satelliten. Zwei verschiedene Verfahren werden beschrieben:

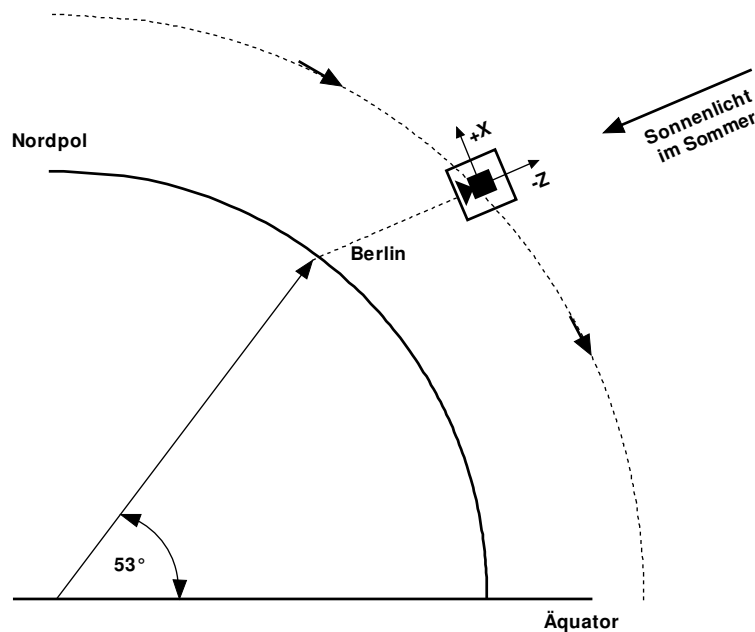
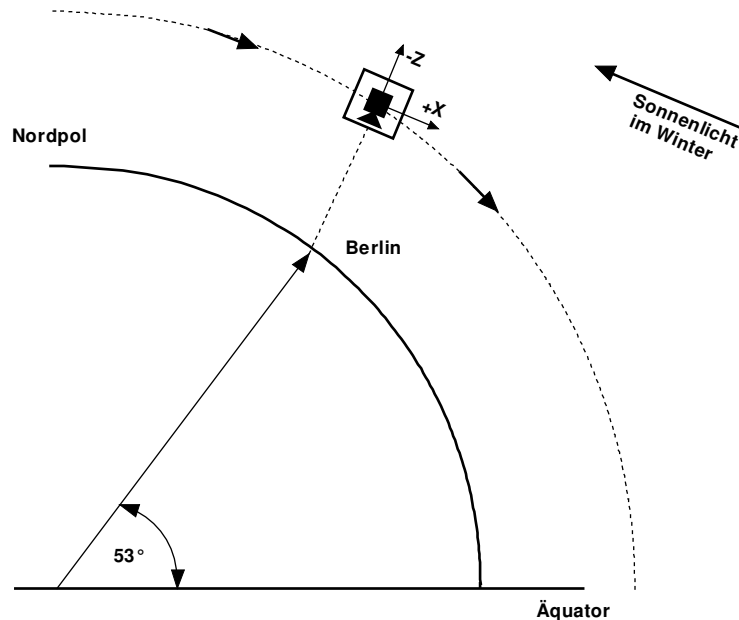


Abbildung 2-1:Vorstabilisierung im Sommer, -Z zur Sonne [1]

Beim ersten Verfahren wird zunächst das der Nutzlastkamera gegenüberliegende Solarpanel (-Z) des Satelliten zur Sonne ausgerichtet. Damit ist der Satellit in einer Achse ausgerichtet. Die zweite Achse wird dadurch festgelegt, dass ein Schwenk um die Bildachse durchgeführt wird, sodass ein senkrechter Vorschub des Live-Videobildes von oben nach unten zu beobachten ist. Mit diesem Verfahren kann man den Satelliten nur auf Ziele vorstabilisieren, die während des ausgewählten Überflugs vom „Schatten“ des Satelliten, also dem Schnittpunkt der Verlängerung des Vektors Sonne-Satellit mit der Erdoberfläche, überstrichen werden. Der Schwenk um die Bildachse ist für die sich anschließende interaktive Lageregelung nicht unbedingt notwendig, da die Joystick- bzw. Mausbewegungen in Drehraten um die körperfesten Achsen des Satelliten übersetzt werden. Die Drehung sorgt lediglich dafür, dass der Bildvorschub später durch ein einziges Drehratenkommando um die Y-Achse des Satelliten kompensiert werden kann. Zusätzlich erleichtert sie dem Operator die

Orientierung. Für den Bodenstationsstandort Berlin ist dieses Verfahren vor allem für den Sommer geeignet (siehe Abbildung 2.1), da ansonsten der Blickwinkel zu flach wird.



**Abbildung 2-2:** Vorstabilisierung im Winter, +X zur Sonne [1]

Für das zweite Verfahren wird im ersten Schritt die +X Achse des Satelliten zur Sonne ausgerichtet. Anschließend wird der Satellit um die X-Achse in Rotation versetzt. Die Rotation wird gestoppt, wenn die Stärke des S-Band-Videosignals ein Maximum erreicht hat. Das S-Band-Signal kann jedoch nur dann als Kriterium herangezogen werden, wenn das Ziel der Vorausrichtung die eigene Bodenstation sein soll. Bei diesem Verfahren ist der Blickwinkel im Winter steiler und somit das Maximum der S-Band Feldstärke deutlicher zu erkennen als im Sommer, daher ist es eher für den Winter geeignet.

Anstelle des S-Band-Signals wurden nach Abschluss der Forschungen zu [1] an der TU-Berlin noch andere Kriterien für die Ausrichtung der zweiten Achse untersucht: So wurde z.B. die +Z Seite auf ein Erdalbedo-Maximum geregelt, oder es wurde versucht, direkt anhand des Live-Bildes zu entscheiden, ob das Zielgebiet bereits erreicht wurde.

Alle beschriebenen Verfahren basieren, wie oben bereits erwähnt, auf der Auswertung der Solarzellenströme. Diese Möglichkeit besteht bei Nachtüberflügen nicht. An die Stelle der Solarzellen muss folglich ein anderer externer Sensor treten. Im Rahmen dieser Forschungsarbeit wurde daher die Nutzlastkamera durch die Entwicklung einer speziellen Software zu einem Sternsensor erweitert, dessen Beschreibung im nächsten Kapitel folgt.

Auch die eigentliche interaktive Regelung ist im Rahmen dieser Arbeit weiterentwickelt worden. Bei dem von Schulz [1] beschriebenen Verfahren schließt sich an die Akquisition die interaktive Steuerung mithilfe einer PC-Maus an. Hierbei wird im Sekundentakt die Mausposition in X und Y in Drehratenkommandos für die ACS-Einheiten der entsprechenden Satellitenachsen umgesetzt.

Im Betrieb hat sich gezeigt, dass sich bei diesem Verfahren Probleme ergeben, sobald die Funkverbindung zum Satelliten leicht gestört ist und somit vereinzelt Steuerbefehle nicht durchdringen. Wenn ein Steuerbefehl den Satelliten nicht erreicht, hat der Operator den Eindruck, er hätte die Maus nicht weit genug ausgelenkt. Um dies zu korrigieren führt er in der Folge eine stärkere Steuerbewegung aus. Kommt dieser, nun zu starke, Steuerbefehl schließlich durch, so führt dies zu einer ungewollt schnellen Drehrate um die entsprechende Satellitenachse. Auf diese Weise verliert der Operator dann das Ziel aus dem Bild. Die Maussteuerung wurde daher im Zuge dieser Veränderung durch eine Tastatursteuerung ersetzt, deren Vorzüge in Kapitel 7 ausführlich beschrieben werden.

## ***2.2 Bestehende Systeme für die Erdbeobachtung bei Nacht***

Mit „Erdbeobachtung bei Nacht“ ist in dieser Dissertation immer Erdbeobachtung bei Nacht im sichtbaren Spektralbereich gemeint, d.h. die Beobachtung von natürlichen und künstlichen Lichtquellen, welche für das menschliche Auge sichtbar sind. Streng genommen umfasst der Begriff der Erdbeobachtung auch Bereiche wie Infrarot- oder Radarbilder, welche tags wie nachts aufgenommen werden können. Sie sollen aber nicht Gegenstand dieser Arbeit sein.

Zunächst ist festzustellen, dass es bis heute keine Erdbeobachtungssatelliten gibt, welche ausschließlich für die Aufnahmen von Nachtbildern konzipiert wurden. Bei Großsatelliten zur Erdbeobachtung handelt es sich vielmehr um Systeme, welche eine Vielzahl unterschiedlicher Sensoren mit sich führen, von denen sich einige auch, aber in der Regel nicht nur, für Nachtaufnahmen eignen.

Ein Beispiel hierfür sind die DMSP-Satelliten des US-Militärs. Diese Satelliten wurden als Wettersatelliten ausgelegt, und verfügen über insgesamt 7 verschiedene Instrumente zur Erdbeobachtung. Eines dieser Instrumente, das Operational-Linescan-System (OLS), dessen Hauptaufgabe in der Beobachtung der Wolkenverteilung liegt, ist für Nachtaufnahmen einsetzbar. Die Aufnahme von Nachtbildern stellt also gewissermaßen eine Zweckentfremdung des Satelliten dar. Ungeachtet dessen dienen die Nachtbilddaten des DMSP als Basis für wissenschaftliche Untersuchungen [2][3] (siehe Kapitel 1).

Es ist wahrscheinlich, dass auch andere, höher auflösende, Satelliten in ähnlicher Weise zweckentfremdet eingesetzt werden könnten. Deshalb ist es schwer, Aussagen

über den Stand der Technik im Hinblick auf die Erdbeobachtung bei Nacht zu treffen. Im Folgenden sollen daher nur beispielhaft zwei bestehende Systeme, von denen veröffentlichte Nachtbilder existieren, beschrieben werden.

Das oben erwähnte Defense-Meteorological-Satellite-Program (DMSP), wurde bereits in den 60er Jahren begonnen. Der erste Satellitenstart im Rahmen dieses Projektes fand bereits 1965 statt. Die DMSP Satelliten befinden sich auf sonnensynchronen polaren Erdumlaufbahnen mit einer Bahnhöhe von 830 km. Die Aufnahme der Bilder erfolgt mittels einer Zeilenkamera, welche, Nadir-ausgerichtet, einen 3000km breiten Streifen abdeckt. Auf diese Weise ist jeder einzelne Satellit in der Lage, zweimal pro



**Abbildung 2-3:** DMSP-Nachtbild Europa/Nordafrika [5]

Tag die ganze Erde abzubilden. Das operationelle System besteht aus mindestens zwei Satelliten, deren Bahnen um 6 Stunden versetzt sind, sodass eine globale Abdeckung alle 6 Stunden gewährleistet ist. Um das System bis heute in Betrieb zu halten, wurden regelmäßig ausgefallene Satelliten durch neue ersetzt. Die Anzahl der insgesamt im Rahmen von DMSP gestarteten Satelliten beläuft sich mittlerweile auf 15, der letzte Start erfolgte 1999 [6]. Die Abbildung 2-3 zeigt einen Ausschnitt des Bildes „Earth at Night“ welches von der NASA [5] veröffentlicht wurde. Es zeigt Europa



und Nordafrika bei Nacht und wurde mit dem OLS Instrument der DMSP-Satelliten aufgenommen. Dieses Instrument zeichnet sich dadurch aus, dass es auch bei sehr schwachen Beleuchtungsverhältnissen Bilder aufnehmen kann. Der Hintergrund hierfür ist, dass das OLS dafür ausgelegt wurde, auch bei Nacht Wolken zu erkennen, welche allenfalls durch den Mond schwach beleuchtet werden. Die Auflösung des OLS beträgt maximal 500m [6].

Die Abbildung 2-4 zeigt Deutschland bei Nacht [7]. Sie wurde aus einzelnen Bildern des Erdbeobachtungssatelliten LANDSAT 5 zusammengesetzt und nachbearbeitet. Das ebenfalls US-amerikanische LANDSAT Programm ist fast ebenso alt wie das



**Abbildung 2-4:** LANDSAT-Nachtaufnahme von Deutschland [7]

DMSP. LANDSAT 1 wurde 1972 gestartet. Die Aufgabe der LANDSAT-Satelliten ist das Sammeln von multispektralen Bilddaten der Erdoberfläche für zivile Anwendungen z.B. in den Bereichen Wasser- und Landwirtschaft, Umweltschutz und Kartographie. Daneben stehen die Satelliten auch dem US-Militär zu Aufklärungszwe-

cken zur Verfügung. Die Satelliten befinden sich ebenfalls in einer sonnensynchronen niedrigen Erdumlaufbahn. Die Bahnhöhe betrug bei LANDSAT 1 bis LANDSAT 3 (1972-1983) 917 km, ab LANDSAT 4 (1982) wurde eine Bahnhöhe von 705 km gewählt. Bei den derzeit in Betrieb befindlichen Satelliten beträgt das maximale Auflösungsvermögen 30m (LANDSAT 5) bzw. 15m (LANDSAT 7) [8].

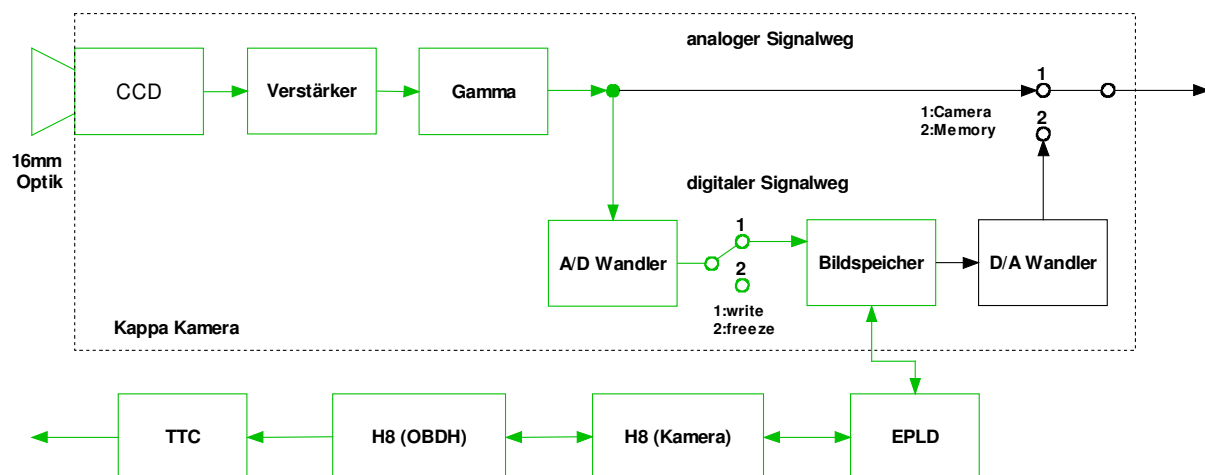
## 3 Entwicklung des Sternsensors

### 3.1 Ausgangssituation

#### 3.1.1 Kamera Hardware

Das Kamerasystem des DLR-TUBSAT wurde ursprünglich so konzipiert, dass es zwei Funktionen erfüllt: Zum einen war der Einsatz als Nutzlastkamerasystem zur Erdbeobachtung vorgesehen, zum anderen sollte es aber gleichzeitig als Sternsensor und somit als Teil des Lageregelungssystems eingesetzt werden können.

Die Nutzlastplattform wurde mit drei optischen Köpfen mit jeweils unterschiedlichen Brennweiten (16mm, 50mm und 1000mm) ausgestattet, um bei der Erdbeobachtung die interaktive Zielfindung zu erleichtern. Dem 16mm- und dem 50mm-Kopf



**Abbildung 3-1:** Blockdiagramm des Kamerasystems an Bord von DLR-TUBSAT [9]

war hierbei die Rolle als Vorfeldsensoren zugeordnet, d.h. mit ihnen sollte das Erfassen des Zielgebietes mit der hochauflösenden Optik vorbereitet werden. Im Funktionsmodus als Sternsensor sollten die drei Köpfe eingesetzt werden, um den Sensor gewissermaßen mit drei verschiedene Empfindlichkeitsstufen betreiben zu können

Abbildung 3-1 zeigt eine vereinfachte Darstellung der DLR-TUBSAT Kamera Hardware. Anstelle der drei optischen Köpfe ist nur das 16mm-Objektiv dargestellt. Bei den Komponenten innerhalb der gestrichelten Umrandung handelt es sich um Baugruppen der Kamera die von der Firma Kappa, abgesehen von der räumlichen Trennung von optischen Köpfen und Ausleseelektronik, unverändert übernommen wurden. Der Logikbaustein EPLD und der H8-Mikrocontroller für die Kameraansteuerung



ung wurde im Rahmen der Entwicklung des DLR-TUBSAT Nutzlastmoduls hinzugefügt [4]. Da der Bordrechner (OBDH) und die TTC-Einheit in diesem Kapitel ebenfalls eine wichtige Rolle spielen, sind sie auch dargestellt, obwohl sie nicht Teil der Kamera-Hardware sind. Für die Verwendung der Kameraplattform als Sternsensor ist nur der digitale Signalweg relevant, dieser ist in der Abbildung grün dargestellt.

Die 16mm-Optik stellt mit dem CDD-Chip den Anfang der Kette dar. Die dort ausgelesenen Pixelladungen werden nach dem Durchlaufen eines Verstärkers und der Gamma-Korrektur durch einen A/D-Wandler digital umgesetzt und im Bildspeicher abgelegt. Zwischen A/D-Wandler und Bildspeicher befindet sich ein Schalter, der über das Setzen eines Bits im sogenannten Control-Register der Kamera betätigt werden kann. In Schalterstellung 1 wird der Bildspeicher im Rhythmus der Belichtungszeit überschrieben, in Schalterstellung 2 bleibt der Bildspeicher eingefroren [9]. Über den Logikbaustein kann nun der Kamera-Mikrocontroller auf die Bilddaten zugreifen.

Im ROM des Kamera-Mikrocontrollers wurde vor dem Start eine umfangreiche Sternsensorsoftware implementiert deren genaue Beschreibung bei Steckling [4] nachzulesen ist. Es handelte sich um eine Software, die es ermöglichen sollte, die Kamera als Sensor für relative Lageveränderungen des Satelliten zu verwenden, d. h. auf die Integration eines Sternkatalogs zur Bestimmung der absoluten Satellitenlage wurde verzichtet.

Der Sternsensor erwies sich jedoch aus den im folgenden Unterkapitel näher ausgeführten Gründen im Orbit als nicht funktionsfähig. Es musste daher im Rahmen dieser Forschungsarbeit ein Weg gefunden werden, das Kamerasystem so zu modifizieren, dass es als Sensor für die Vorstabilisierung des Satelliten bei Nacht eingesetzt werden kann. Für diese Modifikation kam nur eine rein auf Softwareveränderungen basierende Lösung in Frage, da die Hardware des im Orbit befindlichen Satelliten nachträglich nicht verändert werden kann.

Anders als bei anderen Mikrocontrollern an Bord (z.B. OBDH und TTC-Rechner) verfügt der Kamera-H8 nicht über einen erweiterten RAM in Form von sogenannten externen Pages. Wie bereits erwähnt, wurde die gesamte Software für den ursprünglich geplanten Kameraeinsatz bereits vor dem Start des Satelliten in den ROM des Rechners eingebrannt. Eine nachträgliche softwaremäßige Erweiterung bzw. Veränderung des Systems, wie sie für diese Dissertation durchgeführt wurde, war im Design nicht vorgesehen. Aufgrund dieser knappen Speichersituation an Bord wurden bei der implementierten Software die Aufgaben zwischen Kamera-H8 und Bodenstationsrechner aufgeteilt. Das Sternbild wird dabei vom Satelliten zur Bodenstation übertragen wo dann die Sternmustererkennung bzw. die Bestimmung der Satellitenlage von leistungsfähigeren Rechnern durchgeführt wird.

### 3.1.2 Ursprüngliche Sternerkennung

Mit dem Begriff Sternerkennung wird in dieser Arbeit das Verfahren bezeichnet, mit dem versucht wird, auf dem CCD-Chip des Sensors echte Sterne von Fehlstellen, oder auch Hot-Spots, zu unterscheiden. Bei den Fehlstellen handelt es sich um, in Fol-

Kamera ausschalten
Kamera und ACS einschalten
Räder und Kreisel initialisieren (\$DC 0)
Satellit inertial stabilisieren (Angle=0 für alle Achsen)
Kameraparameter setzen: Auf 16mm Kopf schalten Verstärkung auf 150 Belichtungszeit auf 5s
10s warten
Bildspeicher einfrieren
Akkustisches Signal an Operator am Boden senden

**Abbildung 3-2:** Struktogramm PHTSI

ge der im Orbit auftretenden Strahlung, geschädigte CCD-Pixel. Der Begriff Sternerkennung ist nicht zu verwechseln mit der, in den folgenden Kapiteln ebenfalls häufig vorkommenden, Sternmustererkennung. Diese bezeichnet die Identifizierung des aufgenommen Sternbilds durch den Vergleich mit einem Sternkatalog.

Bei der ursprünglich implementierten Sensorsoftware wird folgendes Verfahren für die Erkennung der Sterne auf dem CCD-Chip herangezogen: Zunächst wird eine Liste der 20 hellsten Pixel im Bild erstellt, sortiert nach ihrer Amplitude. Zur Unterscheidung eines Sterns von einer Fehlstelle, wird nun unter den 8 Nachbarn jedes Pixels dieser Liste ein weiterer heller Pixel gesucht. Existiert ein solcher heller Nachbarpixel, so wird der Pixel als Stern interpretiert.

Unmittelbar nach dem Start des Satelliten stellte sich aber heraus, dass dieses Kriterium nicht geeignet ist, um im Orbit Sterne von Fehlstellen zu trennen. Die Anzahl der Fehlstellen auf den CCD-Chip nahm sehr schnell zu. Hinzu kam, dass viele dieser geschädigten Pixel eine sehr hohe Amplitude aufwiesen. Innerhalb kurzer Zeit war die Listenkapazität für helle Pixel durch solche Fehlstellen gesättigt. Weiterhin wurde beobachtet, dass die Kameraelektronik beim Auslesen der Zeilen auf scharfe Hell-Dunkel-Übergänge mit einer gewissen Verzögerung reagiert, so dass eine helle Fehlstelle automatisch in Ausleserichtung noch 1 bis 2 helle Nachbarpixel hat (siehe Abbildung 3-3). Durch diese Nachbarn werden die Fehlstellen dann fälschlicherweise als Sterne interpretiert. Für den Sternsensor bedeutet dies wiederum, dass er in dieser Form unbrauchbar ist.

## **3.2 Sternerkennung trotz Strahlenschäden**

### **3.2.1 Aufnahme der Sternbilder**

Um eine neue Software für die Sternerkennung entwickeln zu können, musste zunächst die Möglichkeit geschaffen werden, mit dem Satelliten Sternbilder aufzunehmen. Um dies zu erreichen, wurde ein bestehendes Programm für zeitgesteuerte Sternaufnahmen verändert. Das Ergebnis war das Programm PHTSI dessen Struktogramm in Abbildung 3-2 dargestellt ist und dessen Funktionsweise im Folgenden kurz beschrieben wird:

Zunächst wird die Kamera ausgeschaltet. Dies ist notwendig, falls während eines Überfluges mehr als ein Sternbild aufgenommen werden soll. Ein einmal eingefrorener Bildspeicher lässt sich aufgrund eines Defekts der Kamera nur durch Aus- und Anschalten derselben wieder überschreiben, der reguläre Befehl zum Auftauen des Bildes wird ignoriert. Dann werden Kamera, Reaktionsräder und Kreisel eingeschaltet. Den ACS-Einheiten aller drei Achsen wird ein Sollwinkel von  $0^\circ$  vorgegeben, was den Satelliten inertial stabilisiert. Dann werden die Kameraparameter eingestellt. Dass für die Belichtungszeit ein Wert von 5 s gewählt wurde, hat seine Ursache in einem weiteren Defekt der Kamera: Eingestellte Werte für das unterste der drei Langzeitregister der Kamera (Register 13 [9]) werden von der Kamera ignoriert. Dies hat zur Folge, dass keine Belichtungszeiten zwischen 40 ms und 5 s eingestellt werden können. Der Vorteil dieser langen Belichtungszeit ist wiederum, dass sich auch dunklere Sterne (etwa bis Klasse 6) abbilden lassen. Nach einer Wartezeit von 10 s wird das Bild eingefroren und es wird ein Signal über OBDH und TTC zum Boden gesendet, um den Operator darüber zu informieren, dass das Sternbild aufgenommen wurde.

### **3.2.2 Flächenkriterium und Amplitudenkriterium**

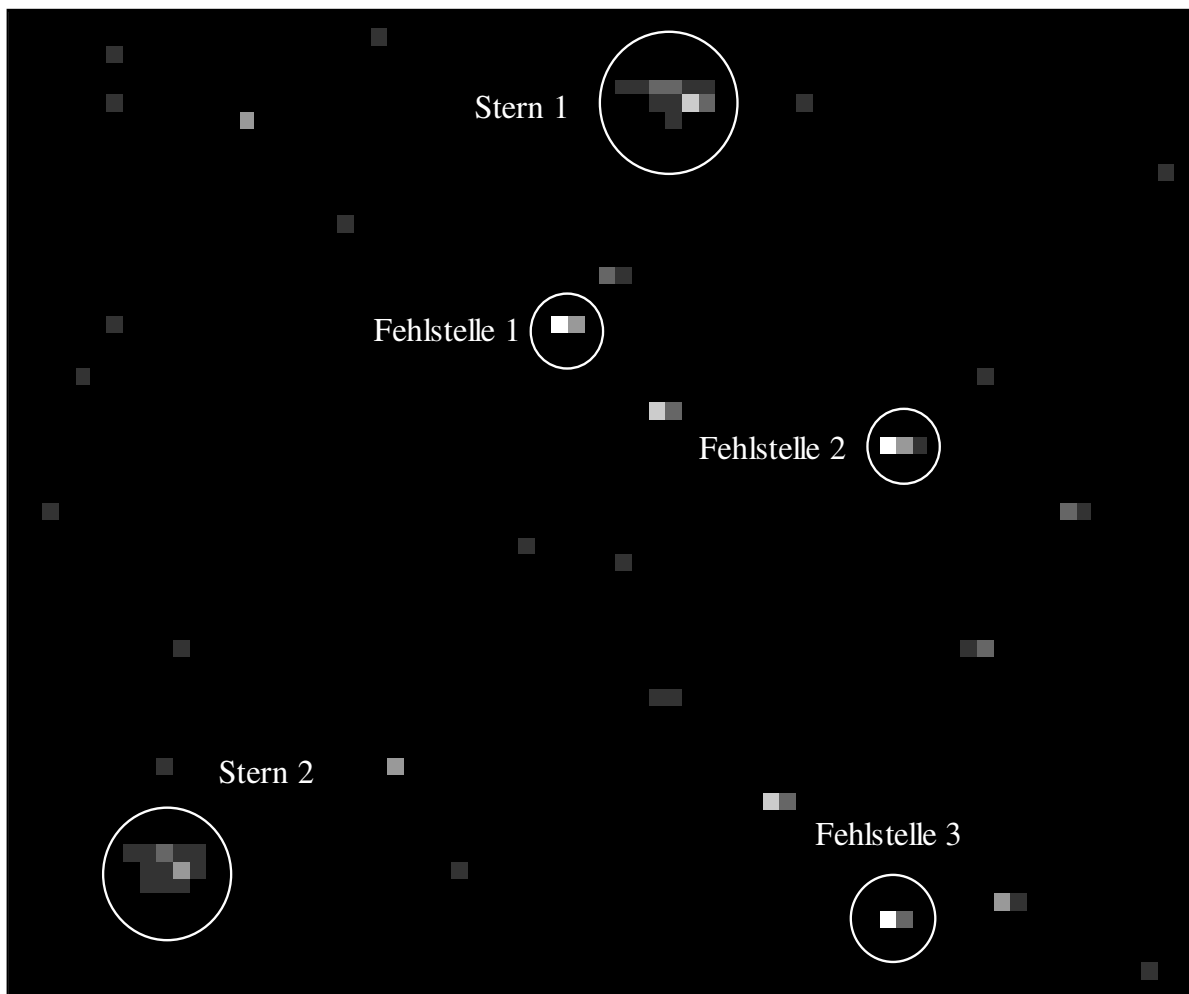
Wie in Unterkapitel 3.1.2 erwähnt, führt die Suche nach den hellsten Pixeln auf dem CCD-Chip nicht zum Ziel, da die Fehlstellen in der Regel gegenüber den echten Sternen in ihrer Amplitude dominieren. Abbildung 3-3 zeigt zur Veranschaulichung einen vergrößerten Ausschnitt aus einem Sternbild, welches fünf Monate nach dem Start des Satelliten aufgenommen wurde.

Im Bild zu erkennen sind zwei Sterne und eine Reihe von Fehlstellen. Die drei markierten Fehlstellen weisen eine höhere Amplitude auf als die Sterne: Der hellste Pixel von Stern 1 hat eine Amplitude von 4, der Maximalwert bei Stern 2 beträgt nur 3. Die Fehlstelle 2 hat hingegen eine maximale Amplitude von 11, Fehlstelle 1 und 3 haben Amplituden von 5 bzw. 7. Diese drei Fehlstellen würden sich folglich in einer nach Amplitude geordneten Rangliste vor die echten Sterne schieben. Ebenfalls im Bild erkennbar sind die hellen Nachbarpixel der Fehlstellen in Ausleserichtung.

Im Rahmen dieser Dissertation wurde nun ein Unterscheidungskriterium für Fehlstellen und Sterne erarbeitet, mit dem trotz der über drei Jahre im Orbit akkumulierten Strahlenschäden noch eine zuverlässige Unterscheidung möglich ist. Ausgenutzt wird dabei die Tatsache, dass das eingesetzte 16mm Objektiv leicht defokussiert ist. Im Gegensatz zu den Fehlstellen bilden sich dadurch Sterne als größere zusammenhängende helle Flächen aus.

Die Abbildungen 3-4, 3-5 und 3-6 zeigen alle dasselbe Sternbild, welches nach ca. 3 Jahren im Orbit aufgenommen wurde. Zur Verbesserung der Darstellung wurden die Bilder invertiert, d.h. Pixel hoher Amplitude sind schwarz dargestellt, Pixel niedriger Amplitude weiß.

Bild 3-4 ist, abgesehen von der nachträglichen Invertierung, unverändert, d.h. die Pixelamplituden entsprechen denen der an Bord des Satelliten ausgelesenen Pixel. Man erkennt, dass das Bild von Pixeln hoher Amplitude übersättigt ist. Es lässt sich mit bloßem Auge jedoch nicht erkennen, ob und wo sich im Bild echte Sterne befinden.



**Abbildung 3-3:** Ausschnitt aus einem Sternbild (Aufnahmezeitpunkt: Oktober 1999)

Bei Bild 3-5 wurde eine Tonwertkorrektur vorgenommen. Die obere Schwelle wurde hierbei von 255 auf 10 heruntergesetzt, d.h. sämtliche Pixel, die eine Amplitude größer als 10 haben, sind nun im Bild schwarz dargestellt. Man erkennt nun mehrere Sterne als Ansammlungen von Pixeln hoher Amplitude. Besonders deutlich zu erkennen ist unten rechts der Stern Mira aus dem Sternbild Walfisch. Links darüber, in der Nähe der Bildmitte, erkennt man den Stern Alrescha, der zum Sternbild Fische gehört.

Diese Maßnahme würde bereits ausreichen für einen eingeschränkten Einsatz des Systems als Sternsensor: Man könnte die vollständigen Sternbilder über die S-Band-Funkstrecke des DLR-TUBSAT zur Bodenstation übertragen und dort mit einer ent-

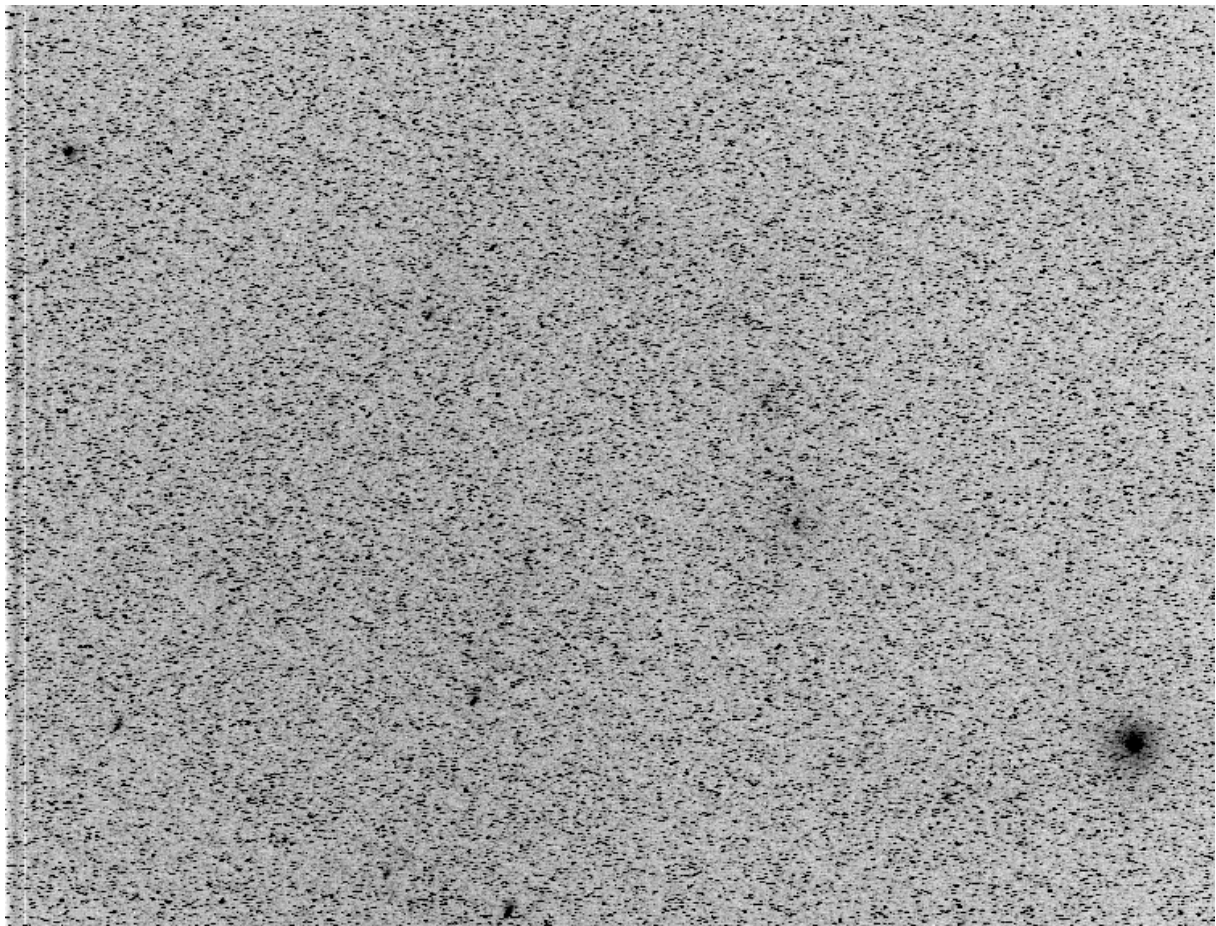


**Abbildung 3-4:** Invertiertes Sternbild nach 3 Jahren im Orbit

sprechenden Software die Sternmustererkennung und Lagebestimmung durchführen. Für die Bildübertragung müsste jedoch gewährleistet sein, dass die S-Band-Antenne, welche sich wie die Nutzlastoptiken auf der +Z-Seite des Satelliten befindet, grob auf die Bodenstation ausgerichtet ist. Dies würde wiederum bedeuten, dass Sternbilder nur tangential an der Erde vorbei aufgenommen werden könnten. Da der Satellit sich zu Beginn des Überflugs jedoch in der Lost-In-Space-Konfiguration befindet, die Satellitenlage also unbekannt ist, ist dies nur sehr mühsam zu erreichen,

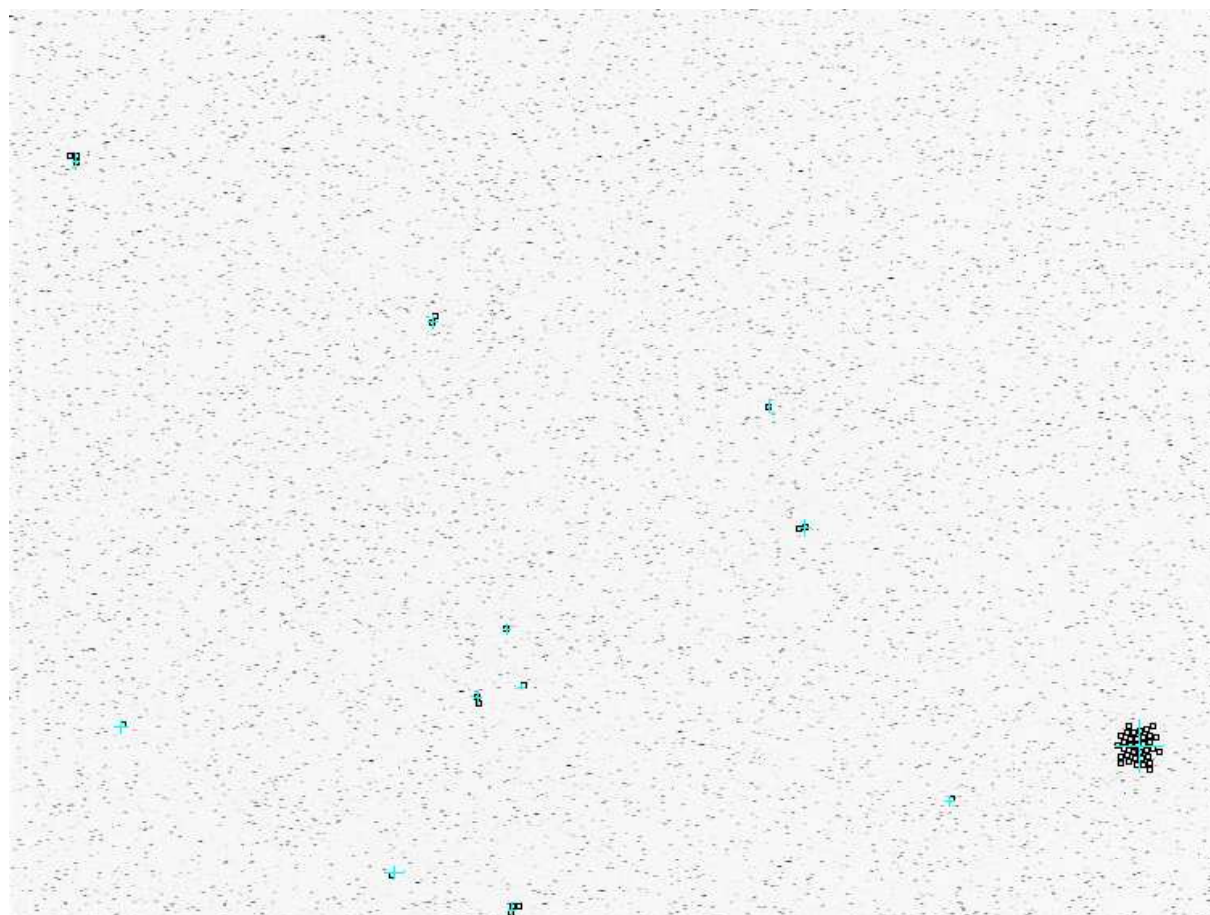
etwa durch eine Kombination aus Trial-And-Error und Auswertung der S-Band Signalstärke. Weiterhin würde das Übertragen des gesamten Bildes viel Zeit in Anspruch nehmen, die anschließend für die Ausführung der Manöver fehlen würde.

Um diese beiden Probleme zu umgehen, wurde für diese Forschungsarbeit ein Algorithmus entwickelt, mit dessen Hilfe sich die Bilddaten an Bord reduzieren lassen. Es kann damit eine schnelle Übertragung des Bildes über die schmalbandige TTC-Funkstrecke, deren Nutzung von der Satellitenlage nahezu unabhängig ist, stattfinden. Die Datenreduktion erfolgt derart, dass zunächst im Bild nach größeren Blöcken heller Pixel gesucht wird. Anstelle des vollständigen Bildes werden dann nur die Koordinaten dieser Blöcke übertragen. Zur Beschreibung der Koordinaten von Objekten auf dem CCD-Chip wird dabei ein im Folgenden als C-System bezeichnetes Koordinatensystem verwendet. Die X-Koordinate legt die Spalte des Blockmittelpunkts auf dem CCD-Chip fest, die Y-Koordinate die Zeile. Der Ursprung des Koordinatensystems befindet sich in der Bildmitte.



**Abbildung 3-5:** Invertiertes Sternbild - tonwertkorrigiert

Bild 3-6 ist während des Entwicklungsprozesses für die Bilddatenreduktionsoftware entstanden. Das Originalbild wurde von einer zunächst für einen PC entwickelten Testsoftware nach flächigen Objekten durchsucht. Der Schwellwert, d.h. die Amplitude ab der ein Pixel als „hell“ interpretiert wird, wurde dabei auf 10 festgelegt. In diesem Fall wurde nach quadratischen Blöcken mit einer Kantenlänge von 4 Pixeln gesucht. Beim Vergleich mit Bild 3-5 erkennt man, dass die Positionen der Blöcke gut mit den hellen Flecken übereinstimmen.



**Abbildung 3-6:** Blocksuche im invertierten Sternbild

Bei dem Verfahren muss darauf geachtet werden, dass die Parameter Schwellwert und Objektgröße nicht auf zu niedrige Werte gesetzt werden, da ansonsten zufällig benachbarte Fehlstellen fälschlicherweise als Blöcke interpretiert werden.

### 3.2.3 Bordseitige Datenreduktion

Das Hauptproblem bei der Entwicklung einer Software für die Datenreduktion an Bord war der geringe zur Verfügung stehende Speicher. Das gesamte Programm inklusive aller Variablen musste im internen RAM des Kamera-Rechners auf insgesamt knapp 1,5 KB untergebracht werden. Aus Tabelle 3-1 geht die Speicheraufteilung des Kamera-Mikrocontrollers nach der Implementierung des Assemblerprogramms zur Bilddatenreduktion (PHDRA) hervor.



Abbildung 3-7 zeigt das Struktogramm welches dem Programm PHDRA zugrunde liegt. Zu Beginn werden die Parameter Höhe, Breite, Versatztoleranz und Schwelle eingelesen, welche die Blöcke charakterisieren, nach denen das Bild durchsucht werden soll. Sie wurden zuvor von der Bodenstation zum Satelliten übertragen und am Ende des RAM abgelegt. Anschließend wird dem Logik-Baustein mitgeteilt, welcher Bereich des CCD Arrays ausgelesen werden soll. Bei dem entwickelten Verfahren wird stets das gesamte erste Halbbild ausgewertet. Das zweite Halbbild kann nicht

	Anfang	Ende	Länge	Bemerkungen
ROM	\$0100	\$F680	62848	
belegter RAM	\$F680	\$F817	407	
PHDRA.BIN	\$F830	\$FB10	736	
PHDRA Parameter	\$FB0C	\$FB10	4	
Blockliste	\$FB10	\$FD08	504	max. 126 Blöcke a 4 Byte
Pixelfeld-Liste	\$FD08	\$FE00	248	max. 62 Pixelf. a 4 Byte
Platz für Stackpointer	\$FE00	\$FE80	128	

**Tabelle 3-1:** Speicheraufteilung Kamera-Rechner

mitberücksichtigt werden, da das EPLD erst nach abgeschlossenen Durchlaufen des ersten Halbbildes mit dem Auslesen des Zweiten beginnen kann. Es müssten also wesentlich mehr Daten im Speicher gehalten werden. Danach wird noch der Schwellwert an das EPLD übergeben, d.h. die Amplitude, ab der das EPLD einen Pixel als sogenannten „Hit“ interpretiert und auf ihn einrastet.

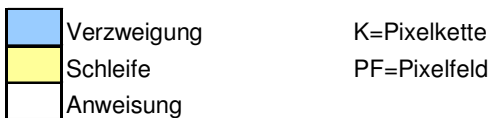
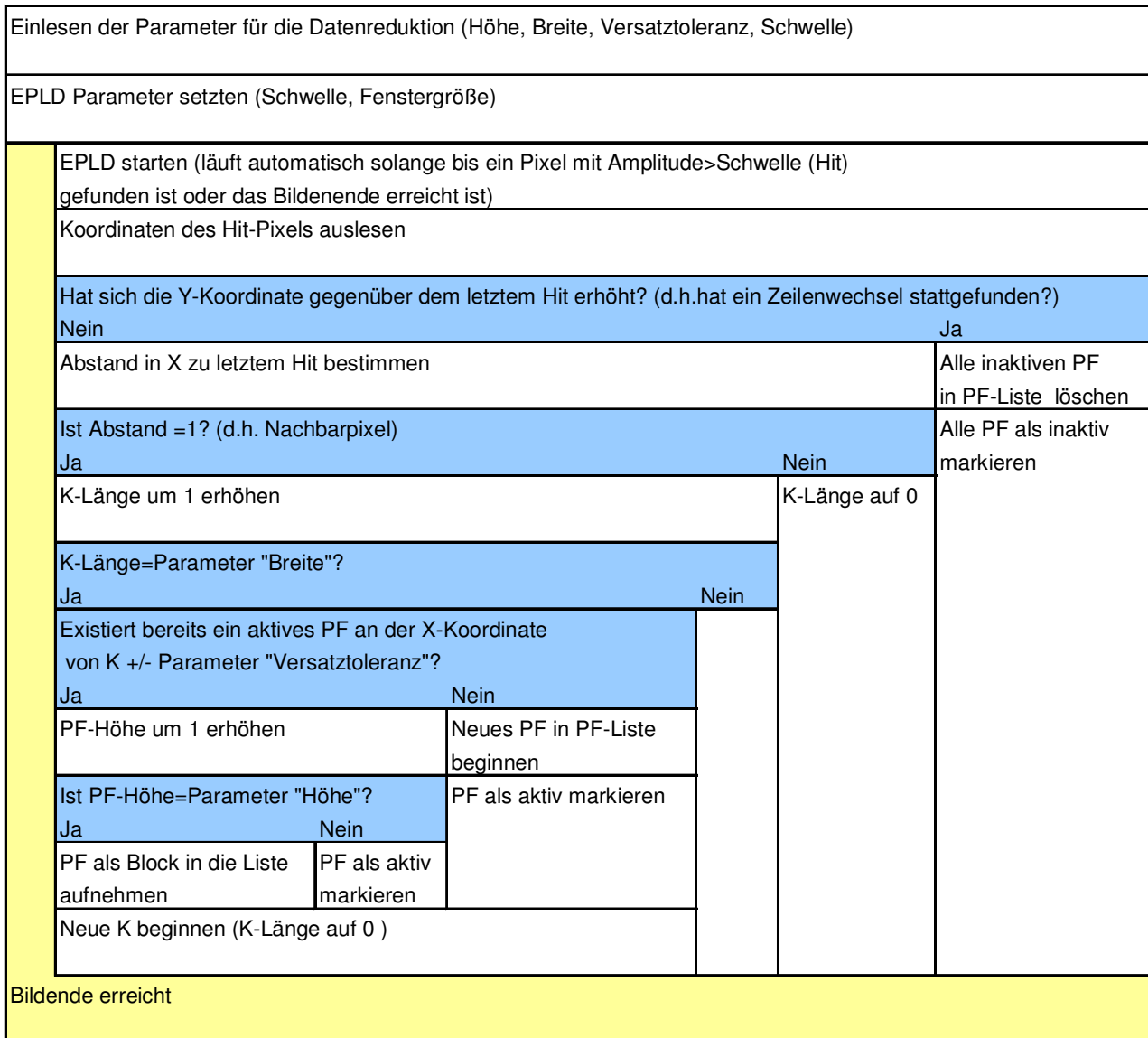
Für das Auslesen des Bildes wird das EPLD gestartet. Es liest fortlaufend Pixel für Pixel. Ist das Zeilenende erreicht, so fährt es mit dem Beginn der nächsten Zeile fort. Erreicht es einen Pixel dessen Amplitude oberhalb des Schwellwertes liegt, so wird der Auslesevorgang angehalten und ein sogenanntes „Hit-Flag“ gesetzt, was dem Kamera-Rechner signalisiert, dass ein heller Pixel gefunden wurde. Dann können die Koordinaten des Pixels ausgelesen werden. Dies ist die einzige zur Verfügung stehende Möglichkeit, um den Bildspeicher auszulesen. Einzelne Pixel gezielt auszulesen ist nicht möglich.

Innerhalb einer Zeile wird zunächst nach zusammenhängenden Ketten heller Pixel gesucht. Die Länge ist dabei über den gleichnamigen Parameter wählbar. Ist eine Kette gefunden, so wird im Speicher ein sogenanntes „Pixelfeld“-Objekt in einer entsprechenden Liste angelegt, das Datenformat eines solchen Objekts ist in Abbildung 3-8 dargestellt.

In den ersten beiden Bytes wird dabei die X-Koordinate der gerade gefunden Pixelkette abgelegt. Das 3.Byte beschreibt die Ausdehnung des Pixelfeldes in Y. Dieser Wert wird für das neue Feld mit 1 initialisiert und erhöht sich, wenn in der nächsten Zeile an ungefähr derselben X-Koordinate, d.h. die Abweichung in X darf nicht grö-



ßer sein, als über den Parameter „Versatztoleranz“ festgelegt, wieder eine Kette gefunden wird (siehe Abbildung 3-9). Im 4. Byte wird die Information abgelegt, dass es sich um ein „aktives“ Pixelfeld handelt. Die Rolle dieser Variable wird im Folgenden erläutert.



**Abbildung 3-7:** Struktogramm PHDRA

Zu Beginn jeder neuen Zeile werden alle Pixelfelder in der Liste als „inaktiv“ markiert. Im Verlauf der Auswertung dieser Zeile wird nun jedes neu begonnene Pixelfeld und jedes bereits bei vorangegangenen Zeilendurchläufen angelegte Feld, dem eine neue Kette zugeordnet werden kann, als „aktiv“ markiert. Letztere werden allerdings nur dann als „aktiv“ vermerkt, sofern ihre Ausdehnung in Y noch nicht den,

über den Parameter „Höhe“ festgelegten, Wert erreicht hat, ansonsten werden diese vollständigen Pixelfelder als sogenannter „Block“ mit X und Y Koordinate wiederum in einer entsprechenden Liste, der „Blockliste“, abgelegt und sie bleiben „inaktiv“. Auch bei Pixelfeldern, denen beim Zeilendurchlauf keine neue Kette zugeordnet werden kann, bleibt das 4.Byte auf dem Wert „inaktiv“ (Beispiel: Abbildung 3-9 c).

Pixelfeld :

1.Byte	2.Byte	3.Byte	4.Byte
X-Koordinate HB	X-Koordinate LB	Höhe	\$AA=Aktiv \$11=nicht aktiv

Block :

1.Byte	2.Byte	3.Byte	4.Byte
X-Koordinate HB	X-Koordinate LB	Y-Koordinate HB	Y-Koordinate LB

**Abbildung 3-8:** Datenformate „Pixelfeld“ und „Block“

Die Einführung dieser Variable hat den Zweck, dass der Auswertungsalgorithmus dadurch mit sehr wenig Speicherplatz auskommt. Nach jedem Zeilenwechsel werden alle inaktiven Elemente der Feldliste sofort gelöscht und der Speicherplatz wird freigegeben. Aus Tabelle 3-1 geht die Aufteilung des für Variablen zur Verfügung stehenden RAM-Bereichs hervor. Er wurde derart aufgeteilt, dass 62 Pixelfelder und 126 Blöcke abgelegt werden können.

Zur Veranschaulichung der in diesem Unterkapitel verwendeten Begriffe und Parameter zeigt Abbildung 3-9 schließlich noch beispielhaft die Suche nach Blöcken für die Parameterwerte Breite=4, Höhe=3 und Versatztoleranz=2.

Für die Realisierung der Datenreduktion an Bord musste noch ein weiteres Problem gelöst werden: Da der Kamera-Mikrocontroller (mit Ausnahme des nicht beschreibbaren ROM-Bereichs) nur über flüchtigen Speicher verfügt, wird der gesamte Speicherinhalt, und damit auch das Programm PHDRA, bei jedem Abschalten der Kamera gelöscht. Da die Kamera zwischen zwei Sternaufnahmen abgeschaltet werden muss, um den Bildspeicher für die neue Aufnahme freizugeben (siehe Kapitel 3.2.1), muss jedes Mal, wenn ein neues Sternbild gemacht werden soll, zunächst das Datenreduktionsprogramm in den Kamera-Speicher übertragen werden. Damit diese Übertragung nicht mühsam von der Bodenstation aus erfolgen muss, befindet sich eine Kopie von PHDRA im EEPROM des OBDH-Rechners, welche mittels eines weiteren Assemblerprogramms (PHLOAD) vom OBDH in die Kamera übertragen wird. Das Programm PHLOAD ist ebenfalls im EEPROM des OBDH untergebracht.

**Beispiel:** Suche nach Blöcken mit den Parameterwerten Breite=4, Höhe=3 und Versatztoleranz=2

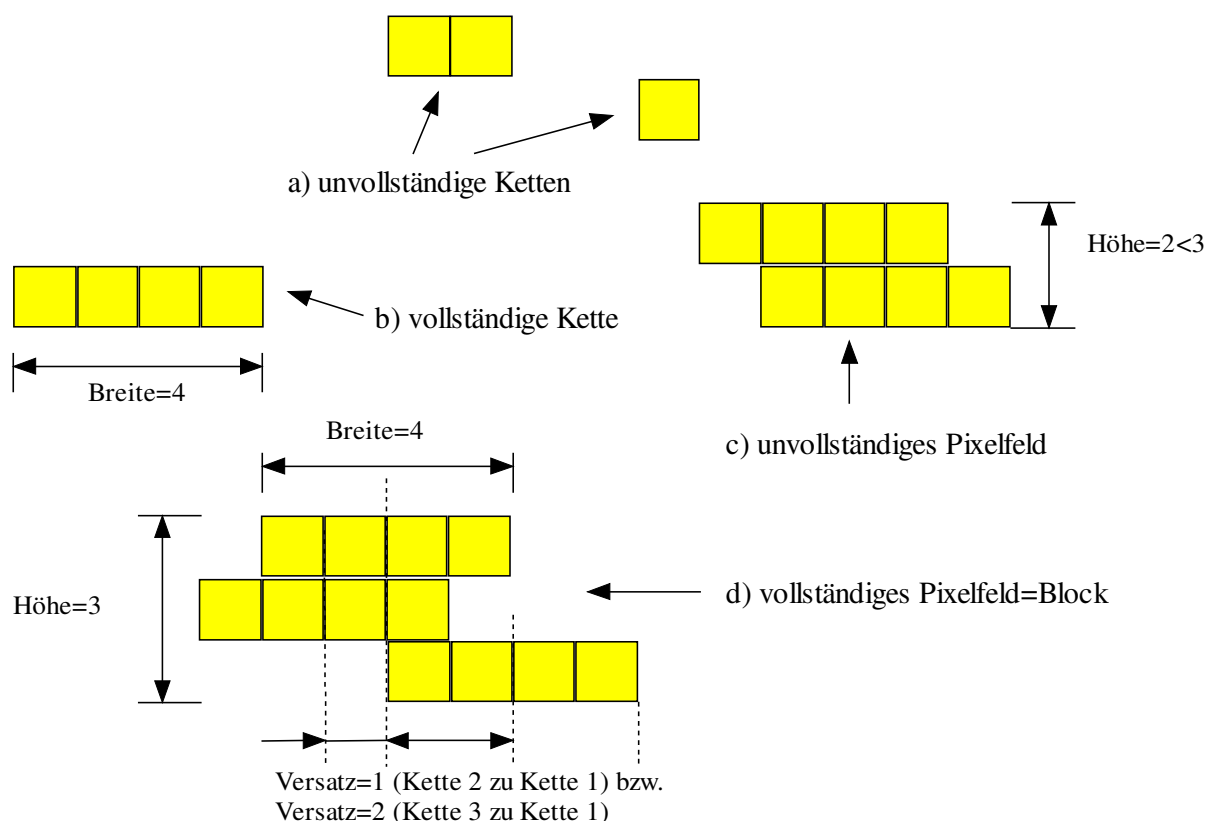


Abbildung 3-9: Beispiel zur Datenreduktion

### 3.2.4 Bodenseitige Datenvisualisierung

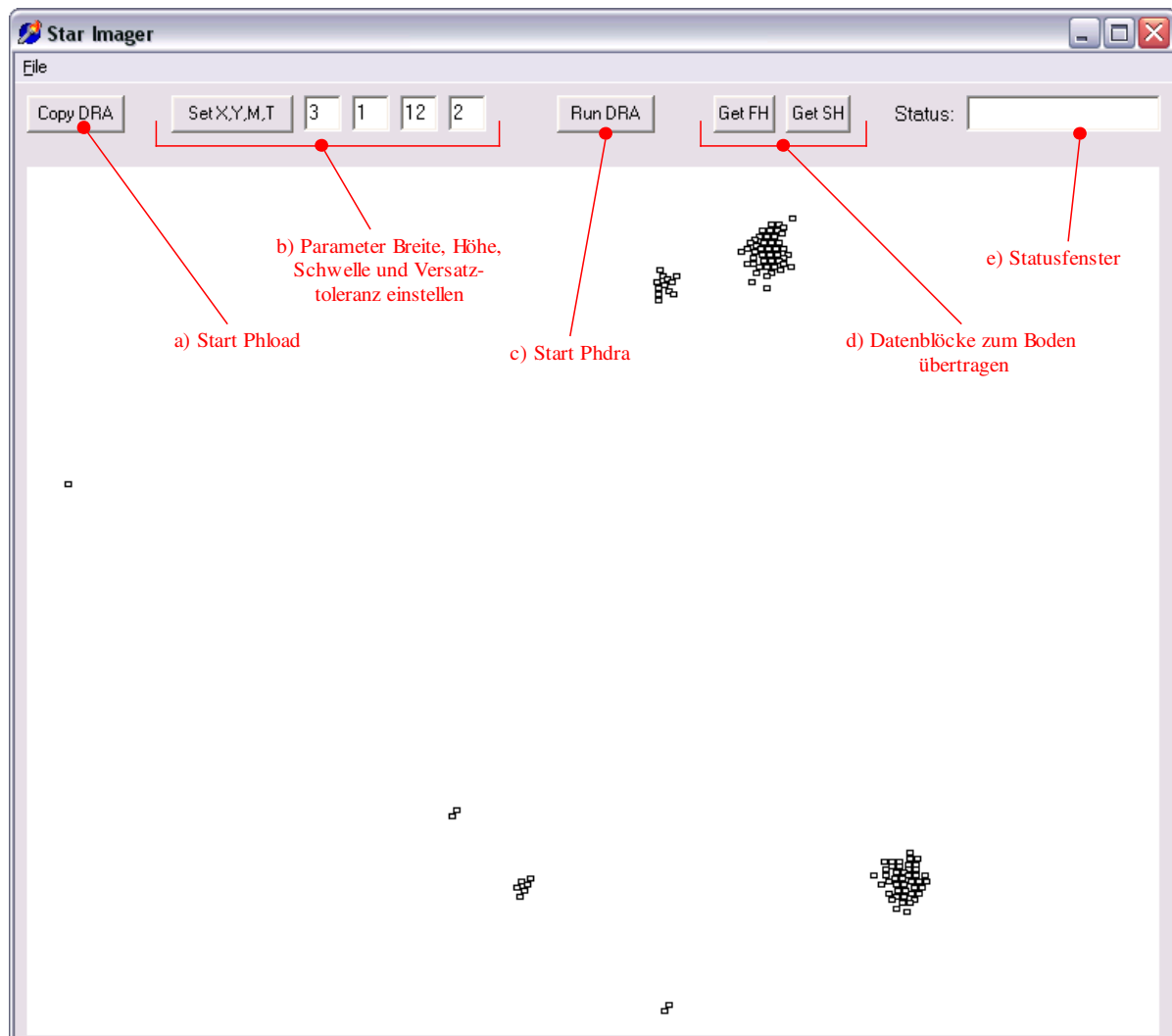
Nachdem die Sternbilddaten an Bord reduziert wurden, werden sie zum Boden übertragen und dort weiterverarbeitet. Der erste Schritt dieser Weiterverarbeitung ist die Visualisierung der Daten, d.h. das Umsetzen der Blockliste in ein Bild. Dieses wieder zusammengesetzte Sternbild dient dann als Grundlage für die Sternmustererkennung bzw. die Lagebestimmung des Satelliten, deren Beschreibung in Kapitel 4 folgt.

Für die Visualisierung wurde das Programm „Star Imager“ entwickelt. Neben der Darstellung des Sternbildes übernimmt es auch die Steuerung der beiden Softwarekomponenten an Bord, PHLOAD und PHDRA. Abbildung 3-10 zeigt die Benutzeroberfläche dieses Programms.

Die Schaltflächen sind so angeordnet, dass sich von links nach rechts eine sinnvolle Befehlsabfolge ergibt. Schaltfläche a) löst das Programm PHLOAD aus, welches, wie im vorangegangenen Unterkapitel beschrieben, das Programm PHDRA vom OBDH in den RAM des Kamera-Mikrocontrollers kopiert.

Die Eingabefelder im Bereich b) ermöglichen die Einstellung der Parameter für PHDRA. Die Reihenfolge der Parameter ist Breite, Höhe, Schwelle und Versatztoleranz. Durch Anklicken der Schaltfläche im Bereich b) werden die eingegebenen Parameter zum Satelliten übertragen.

Durch anwählen der Schaltfläche c) wird daran anschließend Datenreduktion ausgelöst.



**Abbildung 3-10:** Bedienoberfläche von *Star Imager*

Die Schaltflächen bei d) dienen zum Auslesen des 512 Byte großen Datenbereichs. Dieser unterteilt sich in 504 Byte Blockdaten und 8 Byte Parameter- und Statusinformation. Die Übertragung erfolgt in zwei je 256 Byte großen Blöcken, welche unabhängig voneinander abgerufen werden können. Auf diese Weise muss bei Übertragungsfehlern immer nur der fehlerhafte Block erneut geladen werden.

Im Statusfenster e) werden Fehlermeldungen von PHDRA zur Anzeige gebracht. Ist während der Datenreduktion der Speicherbereich der Pixelfeldliste übergelaufen, so

wird die Fehlermeldung „\$EE \$01“ angezeigt. Im Falle eines Überlaufs des Blockspeichers wird „\$EE \$02“ angezeigt. Auf beide Fehlermeldungen kann der Operator dann durch Anpassen der Parameter (z.B. Erhöhung des Schwellwertes) reagieren. Wenn der Algorithmus ohne Fehler durchläuft wird die Bytefolge „\$1A \$1A“ gemeldet.

Die Darstellung der Objekte erfolgt indem zunächst aus dem 8 Byte großen Parameterbereich, der mit dem ersten Block zum Boden übertragen wurde, die Parameter Breite und Höhe ermittelt werden. Anschließend wird für jeden Block aus der übertragenen Liste an den entsprechenden Koordinaten im Ausgabefenster ein Rechteck mit diesen Abmaßen gezeichnet.

Die Prozedur des Anpassens der Parameter und des Herunterladens der Blöcke muss in der Regel mehrmals durchgeführt werden, bis ein verwertbares reduziertes Sternbild entsteht. Damit ein Sternbild verwertbar ist, darf die Anzahl der Blöcke die Listenskapazität nicht überschreiten, gleichzeitig darf sie aber auch nicht zu niedrig sein, da sonst nicht genügend Sterne für die Mustererkennung zur Verfügung stehen.

## 4 Software zur Sternmustererkennung

### 4.1 Sternmustererkennung beim Sternsensor KM 1301

Die Sternmustererkennung ist der nächste Verarbeitungsschritt für das Sternsensorbild. Es werden dabei die im Bild gefundenen Sterne mithilfe eines Mustervergleichs den realen Sternen, deren Daten in Form eines Sternkatalogs vorliegen, zugeordnet. Ist das Sternbild identifiziert, so lässt sich anschließend daraus die Lage des Satelliten berechnen.

Da die für diese Arbeit programmierte Sternmustererkennung dem Verfahren des Sternsensors KM 1301 (einem Kooperationsprojekt zwischen der TU-Berlin und der Firma Kayser-Threde) nachempfunden wurde, soll dieses im Folgenden grob beschrieben werden. Die Beschreibung beschränkt sich auf die Mustererkennung. Für weitergehende Informationen zum System wird auf das Benutzerhandbuch des KM 1301 [10] verwiesen. Im Kern besteht die Mustererkennung aus 5 Unterrouتين, die nacheinander aufgerufen werden:

#### 1. Search Best Ref Star

Diese Routine wählt unter den auf dem CCD-Array gefundenen Sternen einen als sogenannten „Referenzstern“ aus. Hierfür wird die Liste der Sterne durchgegangen. Wird ein Stern gefunden, welcher einen vorher festgelegten Mindestabstand zum Rand des Bildes nicht unterschreitet, so wird dieser zum Referenzstern gemacht.

#### 2. Calc Refstar Distance

Calc\_Refstar\_Distance berechnet für die 10 hellsten der gefundenen Sterne jeweils den Abstand zum Referenzstern. Zusätzlich wird auch der sogenannte „CCD-Richtungswinkel“ jedes Sterns bezüglich des Referenzsterns bestimmt. Dies ist der Winkel zwischen dem Vektor Referenzstern/Stern und der -Y-Richtung (oben) auf dem CCD-Chip.

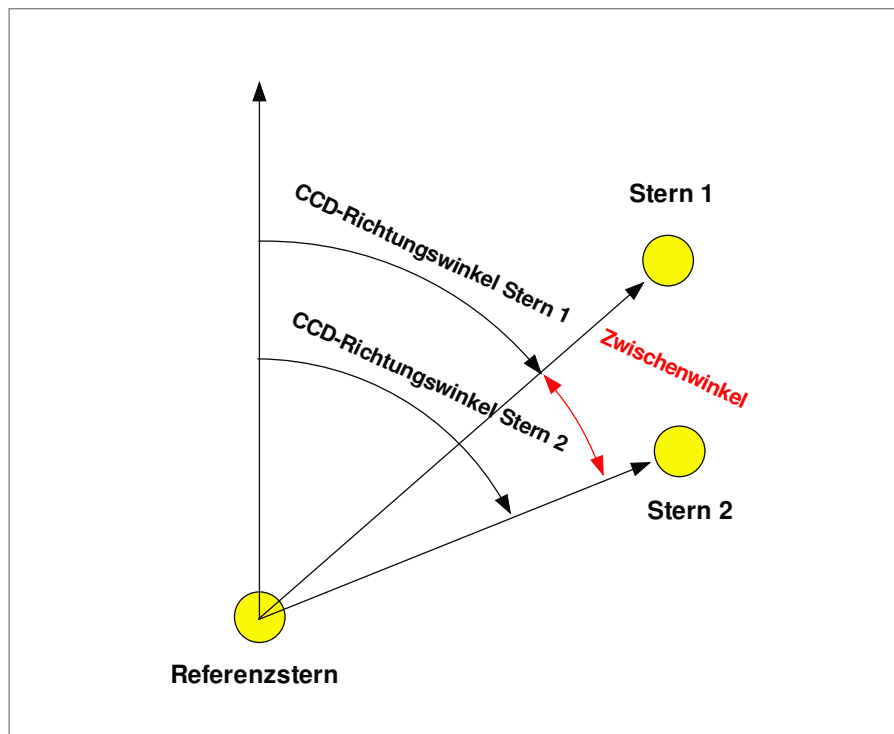
#### 3. Sort Stars

Sort\_Stars sortiert die 10 hellsten Sterne aufsteigend nach ihrem Abstand zum Referenzstern. Die Liste wird abgebrochen sobald der maximal zulässige Abstand, der sogenannte „Musterradius“, überschritten wird.

#### 4. Permut Winkel

Sofern vorhanden, werden für die fünf dem Referenzstern nächstgelegenen Sterne die CCD-Zwischenwinkel jeweils als Differenz ihrer CCD-Richtungswinkel be-

stimmt. Der Zwischenwinkel ist damit der Winkel, den zwei Sterne auf dem CCD-Chip bezüglich des Referenzsterns bilden (siehe Abbildung 4-1). Dieser Winkel wird für jede mögliche Kombination aus zwei CCD-Sternen gebildet, im vorliegenden Fall von 5 Nachbarsternen ergeben sich also 10 Zwischenwinkel.



**Abbildung 4-1:** Richtungswinkel und Zwischenwinkel auf dem CCD-Chip

### 5. Search Muster

In diesem letzten Schritt wird das aufgenommene Sternbild mit einem Musterkatalog verglichen. Der Musterkatalog ist fest im ROM des KM 1301 eingebrannt. Die in ihm abgelegten Sternmuster liegen im selben Format vor wie das in den vorangegangenen Schritten erstellte CCD-Sternmuster. Jedes Muster hat im Zentrum einen Referenzstern. Ihm zugeordnet sind die Entfernungen und Richtungswinkel von 20 Nachbarsternen. Der Richtungswinkel eines Sterns ist hier der Winkel, den der Vektor Referenzstern/Stern mit dem Vektor Referenzstern/Himmelsnordpol bildet. Aus den Richtungswinkeln geht, analog zur Berechnung der Winkel auf dem CCD-Chip, durch Differenzbildung der Zwischenwinkel zweier Sterne hervor.

Der Mustervergleich erfolgt nun so, dass die für das aufgenommene Sternbild errechneten Referenzsternabstände und Zwischenwinkel mit denen der Katalogmuster verglichen werden, d.h. es wird geprüft, ob die Differenz der Werte im Sternbild und im Katalog kleiner als der festgelegte Abstands- bzw. Winkeltoleranzwert ist. Bei dem Vergleich wird eine bestimmte Anzahl von „falschen“ Entfernungen und Zwischenwinkeln toleriert.

Für die im Rahmen dieser Forschungsarbeit programmierte Sternmustererkennung wurde das Prinzip der Mustererstellung und des Mustervergleichs des beschriebenen Verfahrens übernommen. Die verwendeten Muster bestehen ebenfalls aus einem zentralen Referenzstern und einer größeren Anzahl von Nachbarsternen innerhalb eines festgelegten Musterradius. Zur Musteridentifizierung werden auch hier die Abstände der Sterne zum Referenzstern und die Zwischenwinkel von CCD- und Katalogmustern verglichen.

Verändert wurde hingegen der Umfang des Sternkatalogs sowie die Größe der Muster. Der neu erstellte Sternkatalog enthält 5000 Muster bestehend aus jeweils 150 Sternen. Der Musterradius wurde von  $8^\circ$  auf  $20^\circ$  erhöht. Weiterhin wurden im Rahmen dieser Arbeit andere Toleranzkriterien für den Winkelvergleich entwickelt, welche im Abschnitt 4.2.5 vorgestellt werden.

Eine weitere Änderung gegenüber der Vorlage stellt das strengere Mustervergleichskriterium dar: Ein Muster wird nur dann als passend gewertet, wenn ausnahmslos alle Zwischenwinkel und Abstände in diesem vorkommen. Während des Vergleichsprozesses wird daher, sobald zu einem im Sternbild vorkommenden Abstand oder Zwischenwinkel im Muster kein Gegenstück gefunden werden kann, sofort zum nächsten Muster gesprungen. Diese Vorgehensweise spart Rechenzeit bei der Auswertung. Ein Nachteil ist jedoch, dass keine „falschen“ Sterne auftreten dürfen (z.B. Fehlblöcke oder Planeten), da der Algorithmus das Bild dann nicht auswerten kann.

Der größte Unterschied besteht jedoch darin, dass die neuentwickelte Mustererkennungssoftware nicht autonom arbeitet wie der KM1301. Der Operator ist in den Auswertungsprozess des Sternbilds einbezogen. Er muss vor Beginn der Auswertung die zu verwendenden Sterne im Bild markieren und auch abschließend beurteilen, ob das passende Sternbild gefunden wurde. Die genaue Beschreibung der Funktionsweise dieses „halbautomatischen“ Verfahrens zur Mustererkennung folgt im nächsten Unterkapitel.

Für die der Mustererkennung nachgelagerte Lagebestimmung wurde ein eigener Weg auf der Basis des Dreibeinansatzes entwickelt. Die Beschreibung der Lagebestimmung ist Thema von Kapitel 4.2.6.

## **4.2 Halbautomatische Sternmustererkennung**

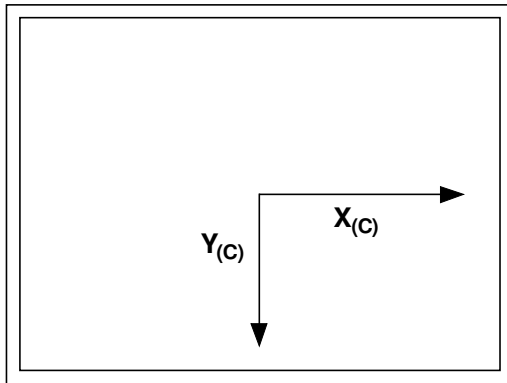
### **4.2.1 Koordinatensysteme**

In diesem Unterkapitel sollen die verschiedenen Koordinatensysteme vorgestellt werden, die in der entwickelten Sternsensorsoftware eine Rolle spielen.

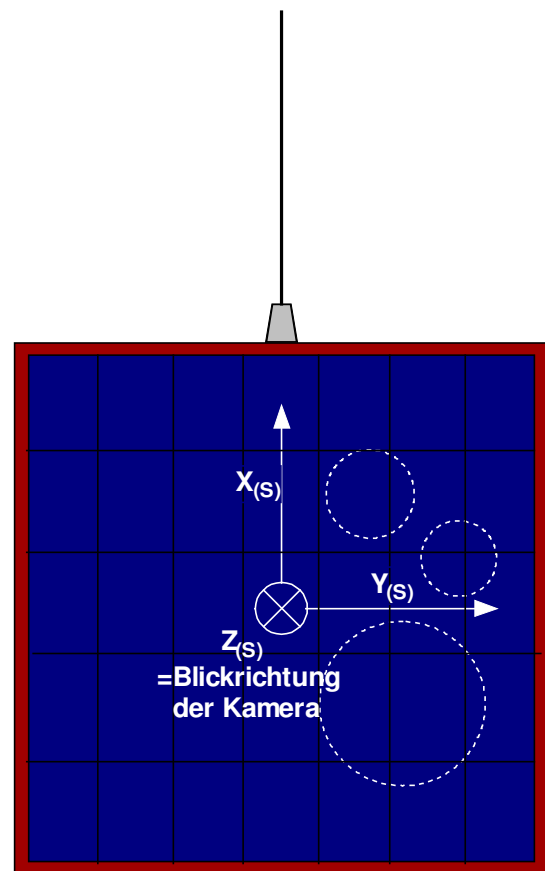


Das in Abbildung 4-2 unter a) dargestellte C-System wurde bereits in Kapitel 3 eingeführt und dient der Positionsbeschreibung von Objekten auf dem CCD-Chip. Es findet Verwendung bei der Umwandlung des reduzierten Sternbildes in ein Sternmuster, das anschließend mit den Mustern des Katalogs abgeglichen werden kann.

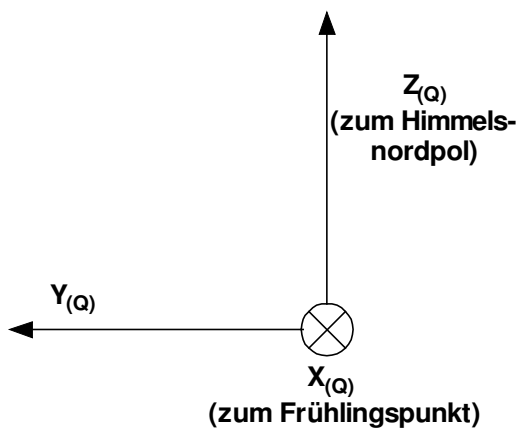
**a) Koordinaten auf dem CCD-Chip (C-System)**



**c) Satellitenfestes Koordinatensystem (S-System)**

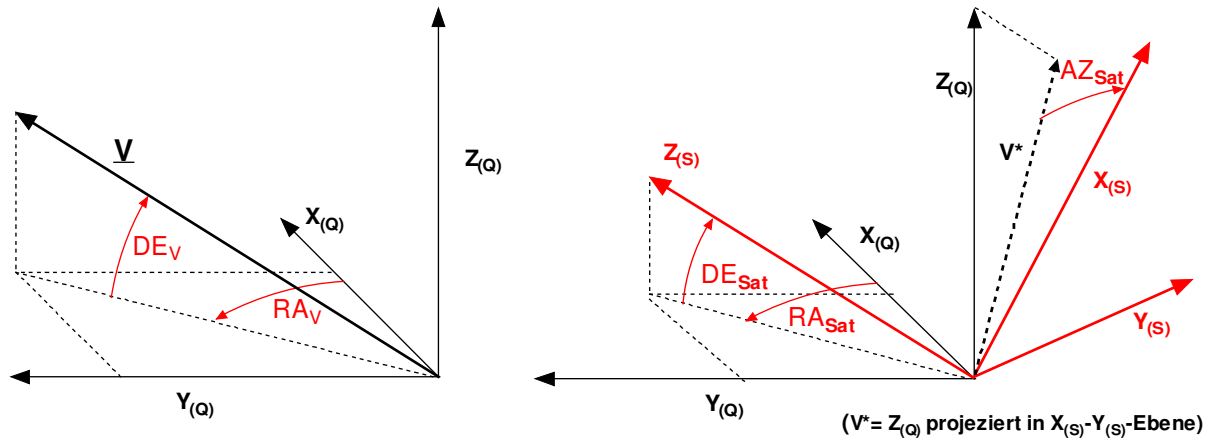


**b) Inertial festes Koordinatensystem (Q-System)**



**Abbildung 4-2:** Koordinatensysteme

Unter b) ist das inertial feste als Q-System bezeichnete Koordinatensystem dargestellt. Dieses Koordinatensystem wird für die Darstellung von Richtungsvektoren, z.B. von Sternen, im Inertialraum benötigt. Die X-Achse weist zum Frühlingspunkt und die Z-Achse zum Himmelsnordpol. Die Y-Achse ergänzt das System zu einem Rechtssystem [11]. Die Darstellung von Vektoren im Q-System erfolgt häufig unter Verwendung der Winkelkoordinaten Rektaszension (RA) und Deklination (DE). Die Rektaszension beschreibt den Winkel, den die Projektion eines Vektors  $V$  in die  $X_q$ - $Y_q$ -Ebene mit der  $X_q$ -Achse bildet. Die Deklination ist der Winkel zwischen dem Vektor  $V$  und der  $X_q$ - $Y_q$ -Ebene. Abbildung 4-3 verdeutlicht dies.



**Abbildung 4-3:** Rektaszension, Deklination und Azimut zur Beschreibung von Vektoren bzw. der Satellitenlage im Inertialraum

Das im Folgenden als S-System bezeichnete satellitenfeste System ist unter c) dargestellt. Die Bezeichnungen der Satellitenachsen wurde aus den DLR-TUBSAT-Projektunterlagen [12] übernommen. Das S-System dient im Zusammenspiel mit dem Q-System zur Lagebeschreibung des Satelliten.

Damit die Beschreibung der Satellitenlage erfolgen kann, muss zunächst definiert werden, für welche Lage des Satelliten (bzw. des satellitenfesten Koordinatensystems) innerhalb des Inertialraums  $RA=DE=AZ=0$  gelten soll. Der Inertialraum wird dabei repräsentiert durch das Q-System. Die „Nulllage“ geht aus Abbildung 4-2 b) und c) hervor. Das satellitenfeste Koordinatensystem ist für  $RA=DE=AZ=0$  innerhalb des Q-Systems wie abgebildet orientiert: Die Kameraachse ( $Z_S$ -Achse) weist zum Frühlingspunkt ( $X_Q$ -Richtung) und die  $X_S$ -Achse weist zum Himmelsnordpol ( $Z_Q$ -Richtung).

Wird die Satellitenlage in Form der Sternwinkel Rektaszension, Deklination und Azimut angegeben, so beschreiben die ersten beiden Winkel die Lage der Kameraachse ( $Z_S$ -Achse) im Q-System und der Azimut die Lage des Satelliten um die  $Z_S$ -Achse. In Abbildung 4-3 rechts ist dies grafisch dargestellt. Die zugehörige Transformationsmatrix lautet:

$$(\_)_{SQ} = \begin{pmatrix} s(AZ)s(RA) - c(AZ)s(DE)c(RA) & -s(AZ)c(RA) - c(AZ)s(DE)s(RA) & c(AZ)c(DE) \\ c(AZ)s(RA) + s(AZ)s(DE)c(RA) & s(AZ)s(DE)s(RA) - c(AZ)c(RA) & -s(AZ)c(DE) \\ c(DE)c(RA) & c(DE)s(RA) & s(DE) \end{pmatrix} \quad (4.1)$$

(verkürzte Schreibweise:  $s=\sin$ ,  $c=\cos$ )

### 4.2.2 Sternauswahl im reduzierten Sternbild

Bei der für diese Dissertation implementierten Sternmustererkennung werden die Sterne im reduzierten Sternbild vom Operator am Boden manuell, d.h. per Mausklick, angewählt. Dieser Weg hat sich als sehr nützlich erwiesen. Ein Operator kann schnell feststellen, ob es sich bei dem aufgenommenen Bild um ein brauchbares Sternbild handelt, oder ob sich z.B. die Erde im Blickfeld befindet und daher eine neue Aufnahme ausgelöst werden muss. Er erkennt beim Betrachten des Bildes unmittelbar, ob es sich bei den hellen Objekten im Bild um Sterne oder um das Licht von Städten auf der Erde handelt: Sterne bilden sich in der Regel als kompakte Blockgruppierung mit scharfer Begrenzung ab, wohingegen Lichtquellen auf der Erde eher ausgefranst wirken. Bei ausreichend steilem Blickwinkel bilden sich Städte zudem, aufgrund der langen Belichtungszeit und der hohen Geschwindigkeit des Satelliten über Grund, länglich ab. Diese Unterschiede springen dem menschlichen Betrachter sofort ins Auge, wohingegen sich eine Automatik damit schwer tun würde.

Mit den gewonnenen Erfahrungen wäre eine vollständige Automatisierung der Mustererkennung technisch zwar realisierbar, jedoch mit erheblichem Programmier- und Testaufwand verbunden, der übertrieben erscheint, da dass das gesamte Vorstabilisierungsmanöver ohnehin von Beginn an von einem Operator begleitet werden muss. So wird er für das koordinierte Auslösen der verschiedenen Programme an Bord des Satelliten genauso benötigt wie für die Anpassung der Parameter des Reduktionsalgorithmusses. Nicht zu vergessen ist auch, dass auf die Vorstabilisierung ja die interaktive Lageregelung folgt, welche auch nicht ohne den Operator auskommt. Es ist also nur konsequent, den Operator auch in die Mustererkennung einzubeziehen und damit Programmieraufwand einzusparen.

Der Operator kann im reduzierten Sternbild bis zu 10 Sterne anwählen. Mit den eingestellten Toleranzparametern, auf welche in den folgenden Unterkapiteln noch eingegangen wird, führt aber bereits die Anwahl von nur fünf Sternen in der Regel zu einem eindeutigen Fit des Sternmusters.

Bei der Auswahl sollten Sterne bevorzugt werden, die aus zwei oder mehr Blöcken bestehen, da einzelne Blöcke mitunter keine echten Sterne sind. Sehr helle und damit entsprechend großflächige Sterne sollten ebenfalls gemieden werden, da sich bei Ihnen das genaue Markieren der Sternmitte schwierig gestaltet.

### 4.2.3 Verwendete Datenstrukturen

Zum besseren Verständnis der entwickelten Software zur Sternmustererkennung werden im Folgenden kurz die eingesetzten Datenstrukturen vorgestellt.

## Datenstruktur für Sterne auf dem CCD-Array : TCCDStar

Variable	Erläuterung	Typ
XS	X-Koordinate	Integer
YS	Y-Koordinate	Integer
PATCP (PATternCounterPart)	Nr. des zugehörigen Mustersterns	Integer
RefDist	Entfernung zum Referenzstern	Float

**Tabelle 4-1:** Datenstruktur TCCDStar

Die Software wurde in der Programmiersprache Delphi geschrieben. Wie auch in vielen anderen Hochsprachen, ist in Delphi die Deklaration von eigenen Datenformaten bzw. Datenstrukturen möglich. Der Hauptnutzen solcher Strukturen ist die Verbesserung der Übersicht über die große Datenvielfalt für den Programmierer.

## Datenstruktur für Sternmuster : TStarPat

Variable	Erläuterung	Typ
RefStar	Referenzstern	TPatStar
Data	zum Muster gehörige Sterne	Array of TPatStar
Size	Anzahl der zum Muster gehörigen Sterne	Integer

**Tabelle 4-2:** Datenstruktur TStarPat

Für die Mustererkennungssoftware werden die zu verwaltenden Daten in zwei Blöcken organisiert: Den ersten Block stellen die Sterne auf dem CCD-Array dar, den zweiten Block der Musterkatalog, mit dessen Hilfe das aufgenommene Sternbild identifiziert werden soll.

Die vom Operator per Mausklick im Sternbild angewählten Sterne werden als Array von CCD-Sternen abgelegt, Tabelle 4-1 zeigt die dafür verwendete Datenstruktur TCCDStar. Jeder Stern ist zunächst charakterisiert durch seine Koordinaten auf dem CCD-Chip. Im Zuge der Bildverarbeitung werden weitere sternbezogene Daten (wie z.B. die Entfernung zum Referenzstern) berechnet, auf die im nächsten Abschnitt näher eingegangen wird. Diese können dann ebenfalls in dem maßgeschneiderten Datenbehälter abgelegt werden. Jeder Stern kann sich also gewissermaßen die für ihn berechneten Daten selbst „merken“.

Der Sternmusterkatalog ist ein Array von Sternmustern. Für die Sternmuster wurde die Struktur TStarPat (siehe Tabelle 4-2) entworfen. Die Struktur umfasst den Referenzstern, die Mustersterne, sowie ihre Anzahl.

Für den Referenzstern und die Mustersterne wurde wiederum die Struktur TPatStar (siehe Tabelle 4-3) erstellt, die alle wichtigen Merkmale der Sterne im Sternmuster umfasst. Es handelt sich bei TStarPat also um eine verschachtelte Datenstruktur.

Datenstruktur für Mustersterne : TPatStar

Variable	Erläuterung	Typ
No	Nummer des Sterns im Sternkatalog	Integer
RA	Rektaszension	Float
DE	Deklination	Float
MA	Magnitude	Float
AzRef	Richtungswinkel bezüglich Referenzstern	Float
CCDCP (CCDCounterPart)	Nr. des zugehörigen CCD-Sterns	Integer
RefDist	Entfernung zum Referenzstern	Float

Tabelle 4-3: Datenstruktur TPatStar

#### 4.2.4 Erstellung des Musterkatalogs

Der für die Sternmustererkennung benötigte Musterkatalog wird vor dem ersten Einsatz der Sensorsoftware einmal erstellt und dann als Datei abgelegt. Das Mustererkennungsprogramm kann dann später die Muster für den Vergleich schnell einlesen, und umfangreichere Berechnungen fallen weg. Ausgangspunkt für die Erstellung ist ein einfacher Sternkatalog. Es handelt sich dabei um eine Liste, in der die, in diesem Fall 5000, hellsten Sterne jeweils mit ihrer Rektaszension, Deklination und Magnitude abgelegt sind.

Bevor mit der Erstellung des Musterkatalogs begonnen werden kann, muss zunächst noch die Mustergröße oder der Musterradius vorgegeben werden. Dieser legt fest, wie weit ein Musterstern vom Referenzstern, der stets den Mittelpunkt eines Musters bildet, maximal entfernt ist. Bei der Software MusterPr wurde ein Radius von 20° gewählt.

Das erste Muster entsteht nun, indem für den ersten Stern der Liste, der automatisch Referenzstern dieses Musters ist, nacheinander der Abstand zu den anderen Sternen der Liste bestimmt wird. Hierfür werden die Himmelskoordinaten, Rektaszension (RA) und Deklination (DE), beider Sterne (nach Abbildung 4-3) in kartesische Richtungsvektoren umgewandelt:

$$V_{Stern} = \begin{pmatrix} \cos(RA) \cdot \cos(DE) \\ \sin(RA) \cdot \cos(DE) \\ \sin(DE) \end{pmatrix} \quad (4.2)$$

Der Abstandswinkel  $d$  ergibt sich dann über das Skalarprodukt der beiden Vektoren:

$$d = \arccos\left(\frac{V_{Stern1} \cdot V_{Stern2}}{|V_{Stern1}| \cdot |V_{Stern2}|}\right) \quad (4.3)$$

Wird ein Stern gefunden, dessen Abstandswinkel zu dem Referenzstern kleiner ist als der vorgegebene Musterradius, dann wird dieser Stern dem Muster zugeordnet, d.h. er wird innerhalb des neu erstellten TStarPat-Objekts in das Musterstern-Array „Data“ übernommen und die Variable „Size“ des Sternmusters wird um eins erhöht. Es wird solange nach weiteren Sternen gesucht, bis entweder alle 5000 Sterne geprüft wurden, oder eine Zahl von 150 Mustersternen erreicht wurde.

Im nächsten Schritt werden alle innerhalb des Musterradius gefundenen Sterne, also die Sterne des Data-Arrays, aufsteigend nach ihrem Abstand zum Referenzstern sortiert.

Anschließend wird für jeden Stern noch der Richtungswinkel bestimmt und in der Variable „AzRef“ abgelegt. Der Vorteil der Berechnung der Richtungswinkel liegt darin, dass später beim Mustervergleich die Winkel zwischen den Sternen sehr schnell durch einfache Differenzbildung ermittelt werden können. Für die Bestim-

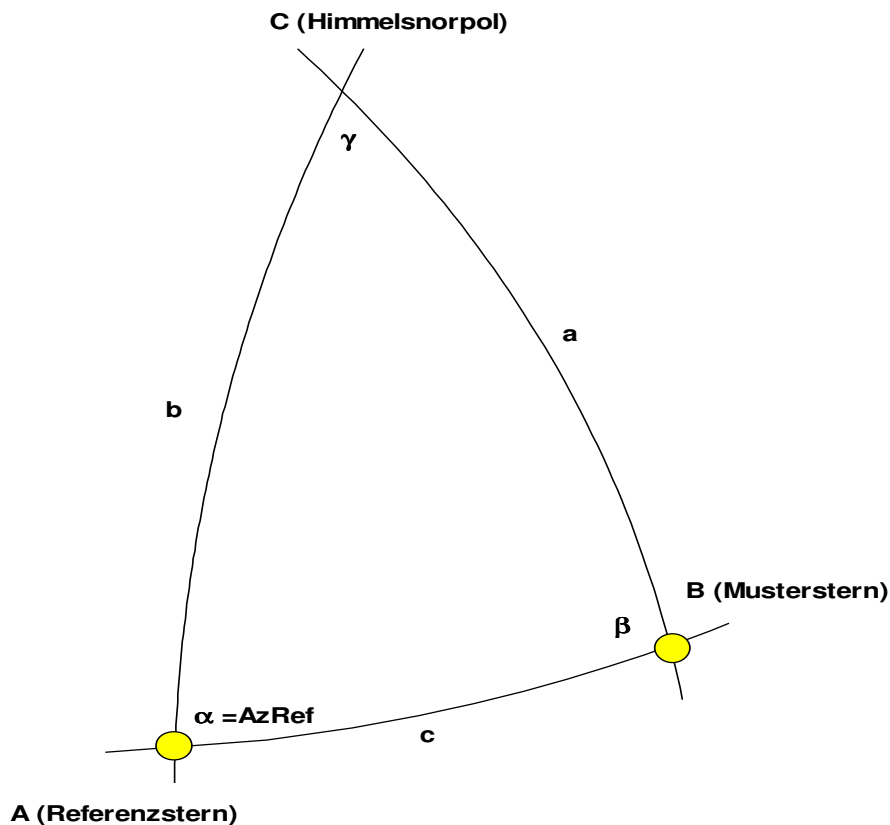


Abbildung 4-4: Richtungswinkel „AzRef“

mung des Richtungswinkels wird der Seitenkosinussatz der sphärischen Trigonometrie herangezogen. Für das in Abbildung 4-4 dargestellte sphärische Dreieck gilt:

$$\cos a = \cos b \cdot \cos c + \sin b \cdot \sin c \cdot \cos \alpha \quad (4.4)$$

Durch Umstellung der Gleichung ergibt sich für den vorläufigen Richtungswinkel  $\alpha^*$ :

$$\alpha^* = \arccos \frac{\cos a - \cos b \cdot \cos c}{\sin b \cdot \sin c} \quad (4.5)$$

Wobei sämtliche Seiten des Dreiecks bekannt sind:

$$a = 90^\circ - DE_{\text{Musterstern}} \quad (4.6)$$

$$b = 90^\circ - DE_{\text{Referenzstern}} \quad (4.7)$$

$$c = d_{\text{Referenzstern} - \text{Musterstern}} \quad (4.8)$$

Um auch für Richtungswinkel zwischen  $180^\circ$  und  $360^\circ$  richtige Ergebnisse zu erhalten, muss zuletzt noch eine Fallunterscheidung durchgeführt werden:

Für  $RA_{\text{Musterstern}} < RA_{\text{Referenzstern}}$ :

$$\alpha = \alpha^* \quad (4.9)$$

Für  $RA_{\text{Musterstern}} > RA_{\text{Referenzstern}}$ :

$$\alpha = 360^\circ - \alpha^* \quad (4.10)$$

Ist das erste Sternmuster fertiggestellt, so wird mit dem zweiten Stern des Sternkatalogs als Referenzstern das nächste Muster erstellt. Auf diese Weise wird mit allen Sternen der Liste verfahren, sodass sich am Ende ein Katalog mit 5000 Sternmustern ergibt.

## 4.2.5 Mustererkennung

Für die Mustererkennung, d.h. für den Vergleich des vom Operator markierten Sternbilds mit dem Musterkatalog wurde die Software MusterPr entwickelt. Abbildung 4-5 zeigt das zugehörige Struktogramm.

Im ersten Schritt muss unter den CCD-Sternen ein Referenzstern ausgewählt werden. Beim KR 1301 wird bei der Auswahl darauf geachtet, dass sich der Referenzstern nicht zu nah am Bildrand befindet, damit ein möglichst großer Umgebungsbereich für den Mustervergleich zur Verfügung steht. Dies ist bei dem Programm MusterPr nicht erforderlich, da der Musterradius, der bei der Erstellung des Musterkatalogs gewählt wurde, etwa der Breite des Sichtfeldes der 16mm Optik entspricht. Dies ermöglicht den Mustervergleich auch dann noch, wenn sich nur am rechten und linken

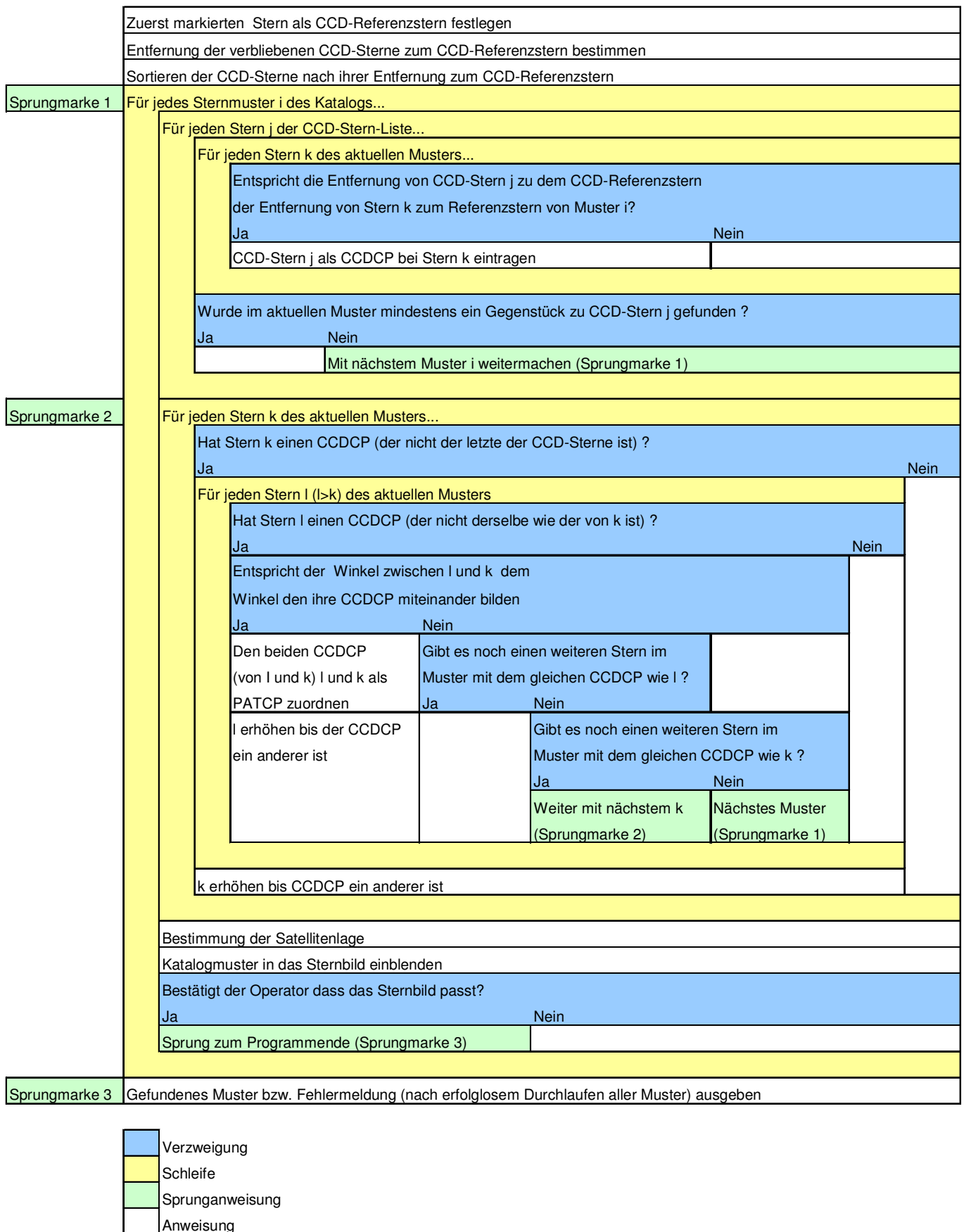


Abbildung 4-5: Struktogramm MusterPr



Bildrand Sterne befinden. Es wird daher der zuerst markierte Stern automatisch als Referenzstern übernommen.

Anschließend werden die Entfernungen der anderen CCD-Sterne zu dem Referenzstern berechnet. Die folgende Formel berechnet den Abstand des  $i$ -ten Sterns zum Referenzstern auf dem CCD-Chip:

$$d_{ccd\_i} = \sqrt{\left(\left(x_{Stern\_i} - x_{Refstern}\right) \cdot k_x\right)^2 + \left(\left(y_{Stern\_i} - y_{Refstern}\right) \cdot k_y\right)^2} \quad (4.11)$$

Dabei beschreiben die Koeffizienten  $k_x$  und  $k_y$  den Winkel, dem ein Pixel in X- bzw. Y-Richtung entspricht. Die Werte von  $0.0292^\circ/\text{Pixel}$  und  $0.0282^\circ/\text{Pixel}$  wurden experimentell anhand von identifizierten Sternbildern ermittelt. Die unterschiedlichen Werte ergeben sich, da die Pixel des CCD-Chips nicht quadratisch sind. Die Pixelabmessungen betragen  $8,6 \mu\text{m} \cdot 8,3 \mu\text{m}$  [4]. Bei dem Sternbild auf dem CCD-Chip handelt es sich um die Projektion eines Ausschnitts der Himmelskugel, einer zweifach gekrümmten Fläche, auf eine Ebene, was eine Verzerrung des Bildes zur Folge hat.

Das Gesichtsfeld der 16mm-Optik ist mit  $22,0^\circ \cdot 16,8^\circ$  jedoch ausreichend klein, so dass diese Verzerrung gegenüber anderen Fehlern<sup>1</sup> in den Hintergrund tritt. Es werden daher die Werte  $k_x$  und  $k_y$  für den gesamten Chip als konstant angenommen. Nach der Bestimmung sämtlicher Entfernungen, werden die Sterne aufsteigend nach ihrem Abstand zum Referenzstern sortiert mithilfe eines Bubblesort-Algorithmusses.

Anschließend werden die für den späteren Winkelvergleich benötigten CCD-Richtungswinkel der Sterne bezüglich der  $-Y$ -Richtung des Chips, welche senkrecht nach oben weist (siehe Abbildungen 4-1 und 4-2), berechnet. Vereinfachend erfolgt diese Berechnung unter Anwendung von ebener anstelle von sphärischer Trigonometrie. Wiederum sind die hierdurch entstehenden Fehler vernachlässigbar gegenüber der Ungenauigkeit beim Markieren der Sterne. Für die Bestimmung des Richtungswinkels jedes Sterns werden zunächst Sinus und Kosinus des Winkels nach folgenden Formeln berechnet:

$$\cos \alpha_{ccd\_i} = \frac{(Y_{Refstern} - Y_{Stern\_i}) \cdot k_y}{d_{ccd\_i}} \quad (4.12)$$

$$\sin \alpha_{ccd\_i} = \frac{(X_{Stern\_i} - X_{Refstern}) \cdot k_x}{d_{ccd\_i}} \quad (4.13)$$

Durch den Einsatz der Arctan2-Funktion ergibt sich daraus der gesuchte Richtungswinkel.

---

<sup>1</sup> Den Hauptfehler stellt das Markieren der Sternmittelpunkte in dem durch die Datenreduktion stark vergrößerten Sternbild dar.

Nun folgt die Suche nach einem passenden Muster im Katalog. Dafür durchläuft die Hauptzählschleife sämtliche Muster des Katalogs. Innerhalb der Hauptschleife erfolgt der Mustervergleich in zwei Stufen:

In der ersten Stufe werden zunächst nur die Sternabstände zum Referenzstern verglichen. Mittels einer Zählschleife wird für die CCD-Sterne nacheinander geprüft, ob es im aktuellen Muster einen oder mehrere Sterne mit dem gleichen Abstand zum Referenzstern gibt. Die Bedingung lautet:

$$\left| d_i - d_{ccd\_i} \right| \leq 2 \cdot k_x \cdot T_{pix} \quad (4.14)$$

$T_{pix}$  ist ein Toleranzwert, welcher in Pixeln angibt, wie weit die Markierung von der tatsächlichen Mitte eines Sterns abweichen darf. Da die Sterne durch die Datenreduktion zu Gruppen von groben Blöcken werden, deren exakter Mittelpunkt schwer zu ermitteln ist, darf dieser Wert nicht zu klein gewählt werden. Es hat sich in der Praxis gezeigt, dass ein Wert von 3 zu guten Ergebnissen führt. Unabhängig von der tatsächlichen Richtung des Abstandsvektors wird der Einfachheit halber der Koeffizient  $k_x$  für die Bestimmung des Toleranzwertes herangezogen. Da diese Toleranz für beide markierten Sterne gelten soll, wird noch ein Faktor 2 eingebracht.

Bei jedem Musterstern, der diese Bedingung erfüllt, wird der CCD-Stern in der Variable CCDCP, also als Sternbildgegenstück, eingetragen. Sobald für einen CCD-Stern kein Gegenstück gefunden wird, wird der Vergleich mit diesem Muster sofort abgebrochen und zum nächsten Muster gesprungen. Auf diese Weise werden unnötige Rechenschritte vermieden und die Suchzeit nach einem passenden Muster verkürzt.

Wenn zu jedem CCD-Stern mindestens ein Gegenstück gefunden wurde, folgt die zweite Stufe des Mustervergleichs. Es wird nun im Muster für jede mögliche Kombination aus zwei CCD-Sternen nach einem passenden Sternpaar gesucht. Ein Mustersternpaar ist dann „passend“, wenn erstens die beiden Mustersterne den CCD-Sternen im Zuge des Entfernungsvergleichs (über die Variable CCDCP) zugeordnet wurden und wenn sie zweitens denselben Winkel bezüglich des Referenzsterns bilden. Diese Stufe des Vergleichs wurde ebenfalls so implementiert, dass sofort zum nächsten Muster gesprungen wird, sobald für eine Kombination kein entsprechendes Sternpaar gefunden wird.

Dieser Winkelvergleich ist im Struktogramm zwar als nur eine Anweisung angegeben, es stecken jedoch mehrere Einzelschritte dahinter, welche im Folgenden kurz beschrieben werden sollen. Zunächst wird für die Sterne  $i$  und  $j$  der Zwischenwinkel auf dem CCD-Chip als Differenz der CCD-Richtungswinkel bestimmt:

$$\varphi_{ccd\_ij} = \alpha_{ccd\_i} - \alpha_{ccd\_j} \quad (4.15)$$

Anschließend wird auch für die Stern\_i und Stern\_j zugeordneten Mustersterne, deren Richtungswinkel ja bereits zum Zeitpunkt der Katalogerstellung berechnet wurden, der Zwischenwinkel bestimmt:

$$\varphi_{ij} = \alpha_i - \alpha_j \quad (4.16)$$

Die Winkel  $\varphi_{ccd\_ij}$  und  $\varphi_{ij}$  werden nun verglichen. Die Messfehler der X- und Y-Koordinaten führen zu entsprechenden Toleranzen, die beim Vergleich mit den katalogisierten Werten berücksichtigt werden müssen.

Die Standardabweichung einer Funktion  $F(X_1, \dots, X_n)$  berechnet sich aus den Standardabweichungen der Parameter  $X_i$  zu

$$\sigma_F^2 = \left( \frac{\partial F}{\partial X_1} \right)^2 \cdot \sigma_{X_1}^2 + \dots + \left( \frac{\partial F}{\partial X_n} \right)^2 \cdot \sigma_{X_n}^2. \quad (4.17)$$

Für den Winkel zwischen den Sternen i und j aus Abbildung 4-6 ergibt sich

$$\varphi_{ij\_ccd} = \arccos \left( \frac{\underline{V}_{ri} \cdot \underline{V}_{rj}}{|\underline{V}_{ri}| \cdot |\underline{V}_{rj}|} \right) \quad (4.18)$$

oder

$$\varphi_{ij\_ccd}(X_i, Y_i, X_j, Y_j, X_r, Y_r) = \arccos \left( \frac{(X_i - X_r) \cdot (X_j - X_r) + (Y_i - Y_r) \cdot (Y_j - Y_r)}{\sqrt{(X_i - X_r)^2 + (Y_i - Y_r)^2} \cdot \sqrt{(X_j - X_r)^2 + (Y_j - Y_r)^2}} \right) \quad (4.19)$$

Die Parameter werden dabei direkt in Pixeln angegeben. Die Tatsache, dass die Pixel nicht exakt quadratisch sind, kann vernachlässigt werden. Nach einigen elementaren Rechenschritten ergeben sich die partiellen Ableitungen schließlich mit  $d_i = |\underline{V}_{ri}|$ ,  $d_j = |\underline{V}_{rj}|$ ,  $X_i - X_r = X_{ir}$ ,  $Y_i - Y_r = Y_{ir}$ ,  $X_j - X_r = X_{jr}$ ,  $Y_j - Y_r = Y_{jr}$  und  $s = \underline{V}_{ri} \cdot \underline{V}_{rj}$  zu

$$\frac{\partial \varphi_{ij\_ccd}}{\partial X_i} = \frac{s d_i^{-2} X_{ir} - X_{jr}}{\sqrt{(d_i d_j)^2 - s^2}} \quad (4.20)$$

$$\frac{\partial \varphi_{ij\_ccd}}{\partial Y_i} = \frac{s d_i^{-2} Y_{ir} - Y_{jr}}{\sqrt{(d_i d_j)^2 - s^2}} \quad (4.21)$$

$$\frac{\partial \varphi_{ij\_ccd}}{\partial X_j} = \frac{s d_j^{-2} X_{jr} - X_{ir}}{\sqrt{(d_i d_j)^2 - s^2}} \quad (4.22)$$

$$\frac{\partial \varphi_{ij\_ccd}}{\partial Y_j} = \frac{s d_j^{-2} Y_{jr} - Y_{ir}}{\sqrt{(d_i d_j)^2 - s^2}} \quad (4.23)$$

$$\frac{\partial \varphi_{ij\_ccd}}{\partial Xr} = \frac{Xir + Xjr - s(d_i^{-2} Xir + d_j^{-2} Xjr)}{\sqrt{(d_i d_j)^2 - s^2}} \quad (4.24)$$

$$\frac{\partial \varphi_{ij\_ccd}}{\partial Yr} = \frac{Yir + Yjr - s(d_i^{-2} Yir + d_j^{-2} Yjr)}{\sqrt{(d_i d_j)^2 - s^2}} \quad (4.25)$$

Im vorliegenden Fall kann ein einheitlicher Wert  $\sigma_p$  für die Standardabweichung aller Koordinaten angenommen werden.

Für  $\sigma_p$  gilt:

$$\sigma_p = T_{pix} \quad (4.26)$$

Damit lässt sich nun die Standardabweichung der Gesamtfunktion nach Gleichung 4.19 bestimmen. Sie lautet:

$$\sigma_\varphi = \sigma_p \sqrt{\left(\frac{\partial \varphi_{ij\_ccd}}{\partial Xi}\right)^2 + \left(\frac{\partial \varphi_{ij\_ccd}}{\partial Yi}\right)^2 + \left(\frac{\partial \varphi_{ij\_ccd}}{\partial Xj}\right)^2 + \left(\frac{\partial \varphi_{ij\_ccd}}{\partial Yj}\right)^2 + \left(\frac{\partial \varphi_{ij\_ccd}}{\partial Xr}\right)^2 + \left(\frac{\partial \varphi_{ij\_ccd}}{\partial Yr}\right)^2} \quad (4.27)$$

Abschließend ergibt sich folgende Bedingung für den Winkelvergleich:

$$|\varphi_{ij} - \varphi_{ccd\_ij}| < 2 \cdot \sigma_\varphi \quad (4.28)$$

Erfüllen zwei CCD-Sterne diese Bedingung, so werden die zugehörigen Mustersterne in die Variable PATCP eingetragen.

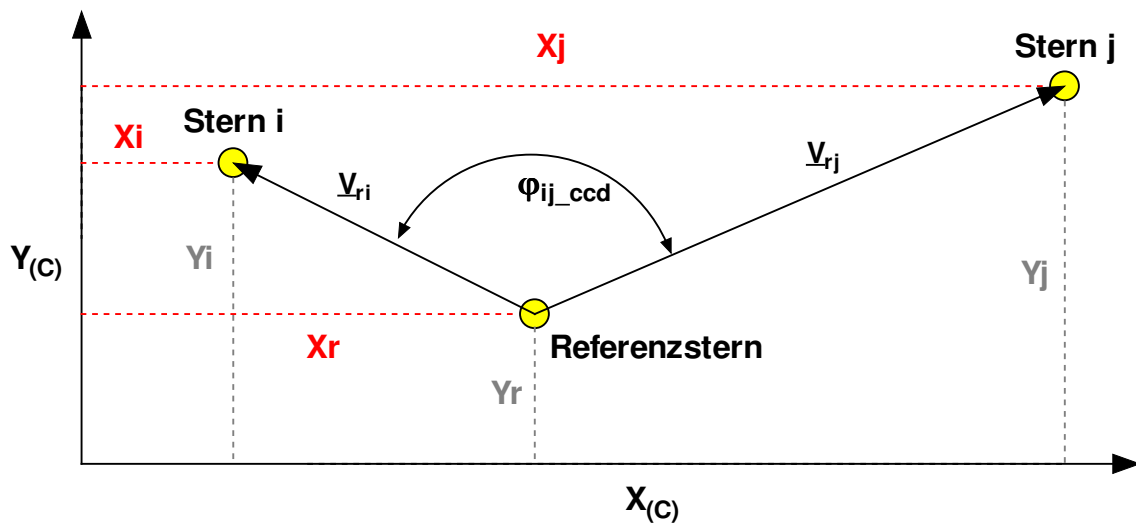
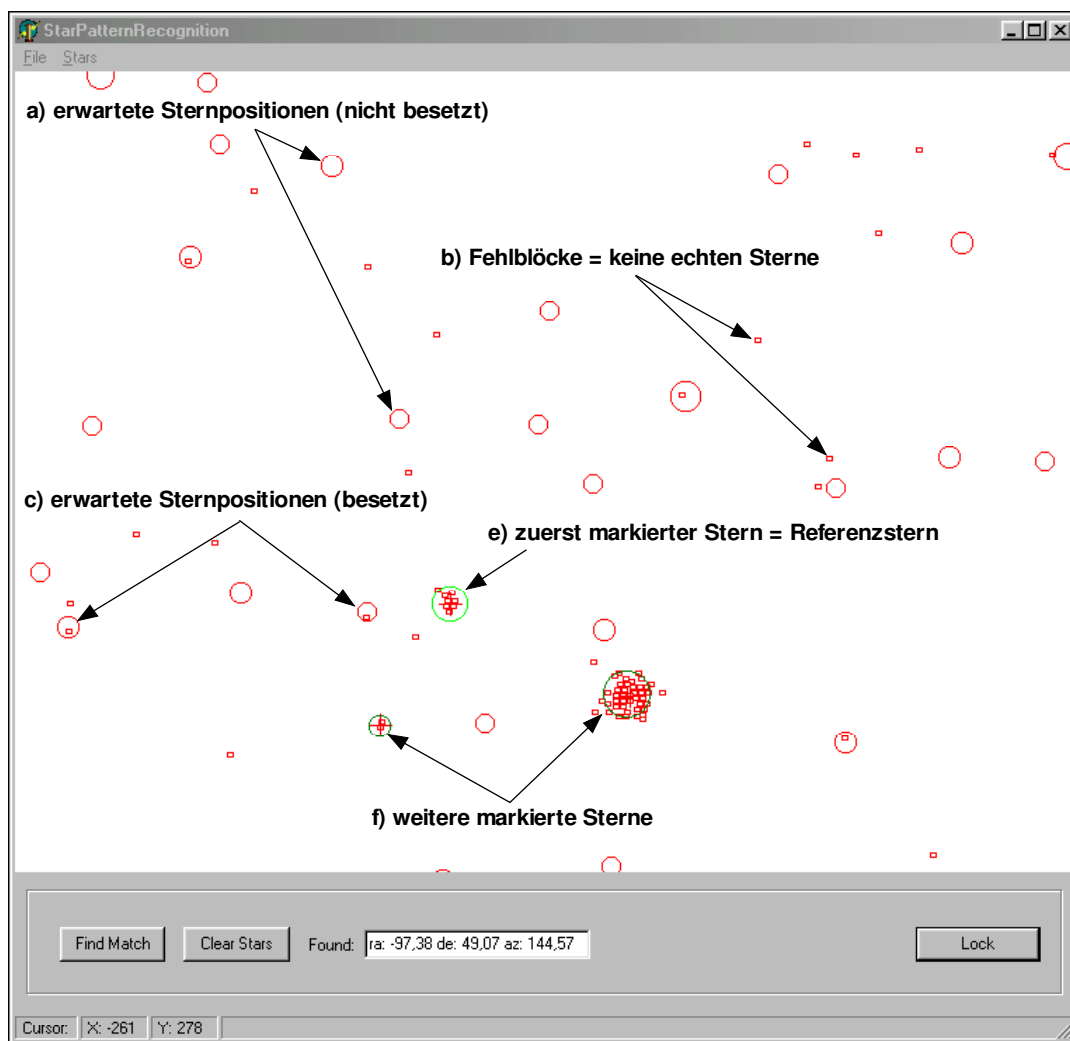


Abbildung 4-6: Bestimmung der Winkeltoleranz

Wenn alle möglichen Winkelkombinationen erfolgreich mit dieser Bedingung geprüft wurden, ist der rechnerische Teil des Mustervergleichs abgeschlossen.

Wurden im Sternbild 5 oder mehr Sterne markiert, so ist das ermittelte Muster in der Regel bereits das passende. Es kommt allerdings häufig vor, dass keine 5 eindeutigen, d.h. aus mehr als einem Block bestehenden, Sterne im Bild ausgemacht werden können. Daher wurde dem Mustervergleich eine weitere Stufe hinzugefügt.

Das ermittelte Sternmuster wird hierbei in das reduzierte Sternbild eingeblendet und der Operator kann entscheiden, ob es sich um das passende Sternbild handelt. Um das Muster richtig im Bild positionieren zu können, muss zuvor jedoch die Lagebestimmung des Satelliten durchgeführt werden, auf welche in Unterkapitel 4.2.6 eingegangen wird. Abbildung 4-7 zeigt beispielhaft ein Sternbild mit eingeblendetem Sternmuster. An jeder Stelle an der sich laut Musterkatalog ein Stern befinden muss-



**Abbildung 4-7:** Einblendung des „erwarteten“ in das „tatsächliche“ Sternbild

te, wird ein Kreis eingezeichnet, wobei die Größe des Kreises die Magnitude des Sterns widerspiegelt. Zur Verbesserung der Übersichtlichkeit wird der Kreis an der Referenzsternposition e) hellgrün dargestellt, die weiteren markierten Sterne f) werden dunkelgrün dargestellt.

Gegenüber der implementierten rechnerischen Auswertung kann der Operator für den Mustervergleich nun zwei zusätzliche Kriterien anwenden.

Er kann erstens überprüfen, ob die Größenverhältnisse der Sterne im Muster und im tatsächlichen Bild übereinstimmen, d.h. ob ein Stern, der laut Musterkatalog eine höhere Magnitude besitzt als ein anderer, im Sternbild auch eine entsprechend größere Fläche hat als dieser. Dies ist in Abbildung 4-7 bei den beiden markierten Sternen f) gut zu erkennen. Der im Musterkatalog als Stern höherer Magnitude eingetragene rechte Stern besteht im reduzierten Sternbild auch aus entsprechend mehr Blöcken.

Zweitens kann der Operator die Objekte, die aus genau einem Block bestehen, mit in die Entscheidung einbeziehen. Je mehr dieser einzelnen Blöcke mit den erwarteten Sternpositionen des Musters übereinstimmen, desto wahrscheinlicher ist es, dass es sich um das passende Muster handelt. Im Beispielbild finden sich fünf solcher Übereinstimmungen, z.B. die unter c) markierten Positionen.

Durch Anwendung dieser beiden Kriterien können auch Sternbilder identifiziert werden, in denen zunächst nur 3 oder 4 eindeutige Sterne markiert werden konnten, und deren rechnerische Auswertung daher mehrere passende Muster liefert. Dafür werden vom Programm nacheinander alle passenden Muster eingeblendet, bis der Operator per Tastendruck bestätigt, dass es sich um das richtige Sternbild handelt.

#### 4.2.6 Bestimmung der Satellitenlage

Im Zuge des im vorigen Abschnitt beschriebenen Mustervergleichs wurde jedem CCD-Stern ein Musterstern über die Variable PATCP zugeordnet. Damit ist der CCD-Stern gewissermaßen identifiziert, d.h. es können nun über sein Muster-Gegenstück alle wichtigen Daten, wie z.B. RA, DE und Richtungswinkel, abgerufen werden. Diese Daten bilden wiederum die Grundlage für die Lagebestimmung.

Die Berechnung der Lage des Satelliten wird innerhalb des Programms MusterPr von zwei Unterrouتين durchgeführt, die im Folgenden beschrieben werden. Die Routine CalcAzimut berechnet den Azimut.

Für die Berechnung wird zuerst für jeden identifizierten CCD-Stern der Azimutwinkel einzeln bestimmt. Der Azimutwinkel  $AZ_i$  eines einzelnen Sterns  $i$  ergibt sich als Differenz seines Richtungswinkels  $\alpha_i$ , der dem Musterkatalog entnommen werden kann, und seines CCD-Richtungswinkels  $\alpha_{ccd\_i}$ , der bereits nach den Gleichungen 4.12 und 4.13 berechnet wurde:

$$AZ_i = \alpha_i - \alpha_{ccd\_i} \quad (4.29)$$

Danach wird aus den einzelnen Winkeln ein gewichteter Mittelwert gebildet. Durch die Gewichtung wird berücksichtigt, dass der Azimutfehler eines Sterns umso gerin-

ger ist, je weiter er vom Referenzstern entfernt ist. Der Azimutwinkel des Satelliten berechnet sich daher als

$$AZ = \frac{\sum_{i=1}^n AZ_i \cdot d_{ccd\_i}}{\sum_{i=1}^n d_{ccd\_i}} , \quad (4.30)$$

wobei  $n$  die Anzahl der markierten Sterne darstellt.

Die Routine CalcChipCenter berechnet die Rektaszension und die Deklination des Sternbildmittelpunkts bzw. des Satelliten. Die Berechnung erfolgt nach dem Dreibeinansatz [11]. Die Vorgehensweise soll anhand von Abbildung 4-8 kurz beschrieben werden. Ausgehend von den Positionen dreier bekannter Sterne, bzw. deren Richtungsvektoren  $\underline{U}_1$ ,  $\underline{U}_2$  und  $\underline{U}_3$ , kann der gesuchte Richtungsvektor des Bildmittelpunkts  $\underline{U}$  bestimmt werden.

Bei der Software MusterPr werden hierfür einfach die ersten drei CCD-Sterne der Liste bzw. deren über die Variable PATCP zugeordneten Katalogsterne herangezogen. Die Rektaszension und Deklination dieser Sterne wird dabei dem Sternkatalog entnommen, die Umrechnung in kartesische Koordinaten erfolgt dann nach Gleichung 4.2.

Für den Dreibeinansatz werden außerdem noch die Abstandswinkel der drei Sterne untereinander ( $\psi_{12}$ ,  $\psi_{13}$  und  $\psi_{23}$ ) sowie die Abstände der Sterne zum gesuchten Bildmittelpunkt ( $\psi_1$ ,  $\psi_2$  und  $\psi_3$ ) benötigt. Da  $k_x$  und  $k_y$  über den gesamten Bildbereich als konstant angenommen werden, ergeben sich die Winkel  $\psi_{12}$ ,  $\psi_{13}$  und  $\psi_{23}$  über Gleichung 4.3 und die Winkel  $\psi_1$ ,  $\psi_2$  und  $\psi_3$  ergeben sich aus

$$\psi_i = \sqrt{(x_i \cdot k_x)^2 + (y_i \cdot k_y)^2} \quad (4.31)$$

Der Vektor  $\underline{U}$  ist darstellbar als

$$\underline{U} = c_1 \cdot \underline{U}_1 + c_2 \cdot \underline{U}_2 + c_3 \cdot \underline{U}_3 \quad (4.32)$$

Da es sich um Einheitsvektoren handelt gilt für die Skalarprodukte der Richtungsvektoren in Abbildung 4-8

$$\underline{U} \cdot \underline{U}_1 = \cos \psi_1 \quad (4.33)$$

$$\underline{U} \cdot \underline{U}_2 = \cos \psi_2 \quad (4.34)$$

$$\underline{U} \cdot \underline{U}_3 = \cos \psi_3 \quad (4.35)$$

und

$$\underline{U}_1 \cdot \underline{U}_2 = \cos \psi_{12} \quad (4.36)$$

$$\underline{U}_1 \cdot \underline{U}_3 = \cos \psi_{13} \quad (4.37)$$

$$\underline{U}_2 \cdot \underline{U}_3 = \cos \psi_{23} \quad (4.38)$$

Durch Einsetzen der Gleichungen 4.32 und 4.36-4.38 in die Gleichungen 4.33-4.35 erhält man das Gleichungssystem

$$\begin{aligned} c_1 + c_2 \cos \psi_{12} + c_3 \cos \psi_{13} &= \cos \psi_1 \\ c_1 \cos \psi_{12} + c_2 + c_3 \cos \psi_{23} &= \cos \psi_2 \quad , \\ c_1 \cos \psi_{13} + c_2 \cos \psi_{23} + c_3 &= \cos \psi_3 \end{aligned} \quad (4.39)$$

durch dessen Lösung man die Richtungskoeffizienten  $c_1, c_2$  und  $c_3$  erhält. Setzt man diese wiederum in Gleichung 4.32 ein, so ergibt sich der gesuchte Vektor  $\underline{U}$ .

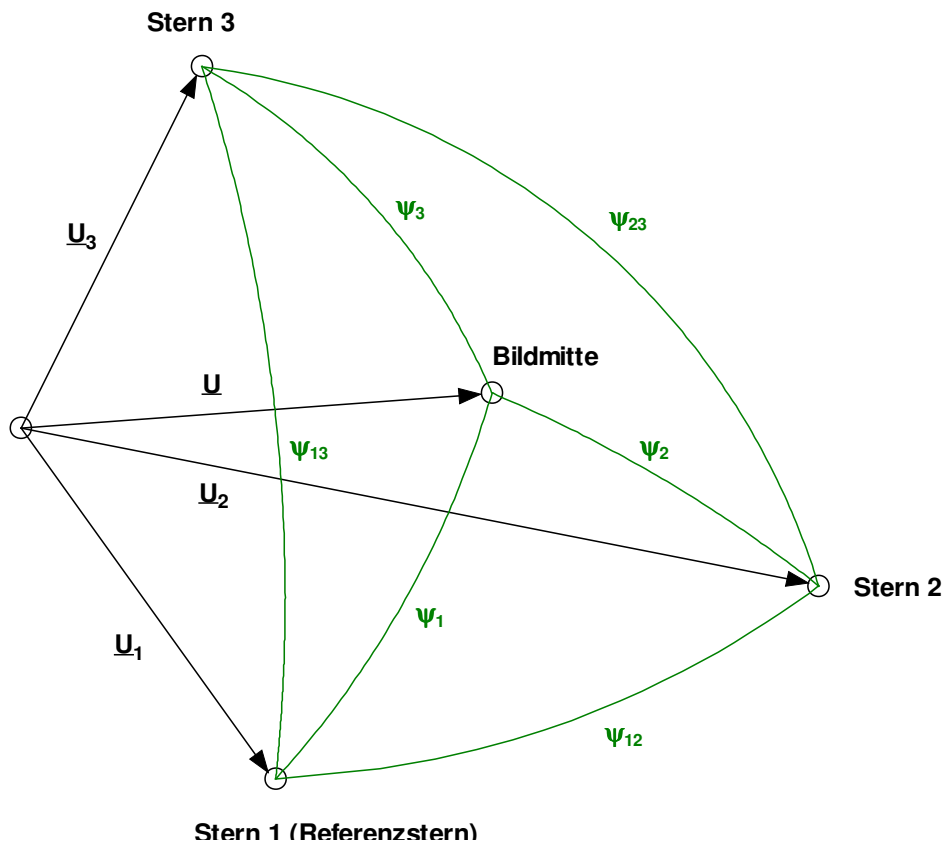


Abbildung 4-8: Dreiecksansatz [11]

Zuletzt müssen noch Rektaszension Deklination aus den Vektorkomponenten von  $\underline{U}$  errechnet werden. Aus

$$RA = \arctan 2(y, x) \quad (4.40)$$

erhält man die gesuchte Rektaszension des Satelliten.



Die Deklination erhält man aus

$$DE = \arcsin\left(\frac{z}{\sqrt{x^2 + y^2 + z^2}}\right) . \quad (4.41)$$

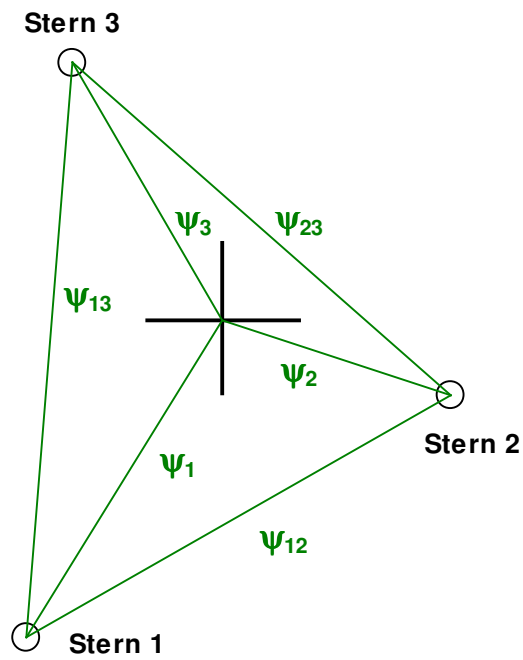


Abbildung 4-9: Dreibeinansatz in der Bildebene

## 5 Simulation und Manöver

### 5.1 Orbit Simulator

#### 5.1.1 Allgemeines

Um den Operator bei der Durchführung des Vorstabilisierungsmanövers zu unterstützen, wurde im Rahmen dieser Forschungsarbeit eine Software entwickelt, welche während des Überflugs in Echtzeit die Bordkameraperspektive, also gewissermaßen den Blick aus dem Satelliten, simuliert. Für die interaktiven Lageregelungsexperimente bei Nacht hat dies den Nutzen, dass der Operator sich während der Anflugphase auf das Ziel an den Küstenlinien, Bergen oder Flüssen dieses simulierten Bildes orientieren kann. Dies wiederum ermöglicht es ihm, die Szene im live übertragenen Nachtvideo zu identifizieren.

Das Simulationsmodell umfasst den Satelliten, die Erde, die Sterne und die Sonne. Da die Simulation nur für den Einsatz während der nur einige Minuten dauernden Vorstabilisierungsphase vorgesehen ist, konnte auf die Modellierung der Störeinflüsse auf die Satellitenlage verzichtet werden. Auch auf die Modellierung des dynamischen Verhaltens der ACS-Einheiten wurde verzichtet, Schwenkmanöver werden in der Simulation vereinfachend mit konstanter Winkelgeschwindigkeit ausgeführt, ansonsten bleibt die Satellitenlage konstant, d.h. der simulierte Satellit driftet nicht.

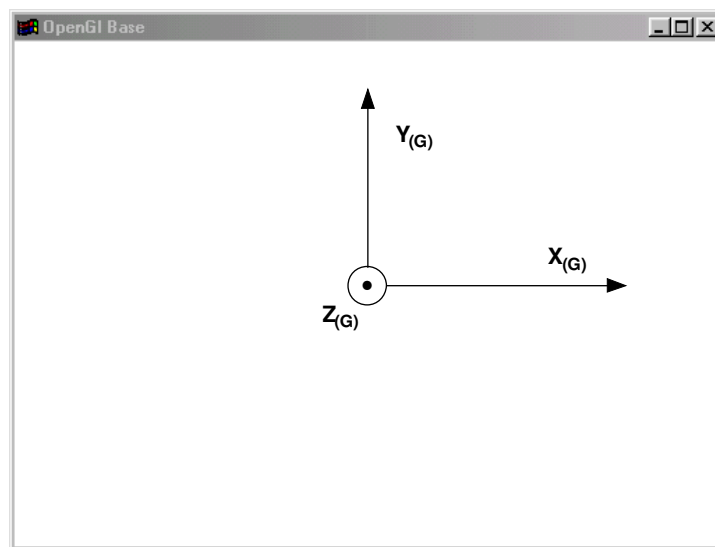
Die grafische Ausgabe der Simulation erfolgt auf OpenGL-Basis. OpenGL steht für Open-Graphics-Library. Es handelt sich dabei um eine Hardware-unabhängige Grafikbibliothek, welche sehr komfortable Funktionen zur Erstellung von 3D-Grafiken bereitstellt.

#### 5.1.2 Modellierung mit Open GL

OpenGL erlaubt das Erstellen und Positionieren von Objekten in einem virtuellen dreidimensionalen Raum. Außerdem ist es möglich, die Betrachterperspektive frei zu wählen und damit quasi eine Kamera beliebig in diesem Raum zu bewegen. Auch der Öffnungswinkel dieser Betrachterkamera lässt sich einstellen. Bei der erstellten Simulation wurde das Gesichtsfeld des 16mm-Objektivs gewählt, um die Bordkameraperspektive möglichst realistisch nachbilden zu können. In diesem Unterkapitel soll die Entwicklung der Simulation mit OpenGL schrittweise erläutert werden. Es werden zu jedem Entwicklungsschritt auch die eingesetzten grafischen Funktionen kurz vorgestellt.

Die Simulation wurde als Windows-basiertes Programm entwickelt, daher erfolgt auch die Ausgabe der OpenGL-Grafik in einem Fenster. Um ein Objekt zu positionieren, beginnt man mit dem Befehl `glLoadIdentity()`. Dieser Befehl bewirkt, dass die Betrachter- bzw. Kameraposition initialisiert wird. Im initialisierten Zustand gilt für das Ausgabefenster das in Abbildung 5-1 dargestellte Koordinatensystem.

Hierbei ist zu beachten, dass sich der Koordinatenursprung in der Bildebene befindet, sodass nur Objekte mit negativer Z-Koordinate sichtbar sind. Objekte mit positiver Z-Koordinate befinden sich gewissermaßen hinter dem Betrachter.



**Abbildung 5-1** Initialisiertes OpenGL-Koordinatensystem

Es ist nun möglich, durch direkte Angabe der Eckpunkte in diesem Koordinatensystem Polygone zu zeichnen, bzw. durch Zusammensetzen mehrerer Polygone komplexere 3D-Objekte zu modellieren.

Sämtliche Objekte, die in diese Simulation eingebettet sind, also Sterne, Sonne und Erde, sind ,zumindest näherungsweise, kugelförmig. Um ein Objekt zu modellieren, dass dem Betrachter als kugelförmig erscheint, ist eine große Anzahl einzelner Polygone notwendig. OpenGL stellt jedoch einen Befehl bereit, der die Erstellung von Kugeln erleichtert, den Befehl `gluSphere()`. Anstelle der einzelnen Polygoneckeppunkte müssen nur noch der Radius sowie die Anzahlen für Längen- und Breitenunterteilungen der Kugel angegeben werden. Je höher diese Werte gewählt werden, desto „runder“ wirkt das Objekt. Die Positionierung erfolgt automatisch im Koordinatenursprung, wobei die Pole der Kugel, also die Schnittpunkte der Längengrade auf der Z-Achse liegen. Soll ein Objekt an einer anderen Position im Raum mit einer anderen Orientierung erstellt werden, so muß zunächst das Koordinatensystem verschoben und rotiert bzw. transformiert werden.

Das Verschieben erfolgt mit dem Befehl `glTranslate()`. Es müssen dabei die Koordinaten des neuen Ursprungs in X,Y und Z im alten Koordinatensystem angegeben wer-

den. Zum Rotieren des Koordinatensystems verwendet man den Befehl `glRotate()`. Übergabewerte sind hier der Drehwinkel, sowie die Drehachse der gewünschten Rotation. Sind auf diese Weise die gewünschte Position und Orientierung des Koordinatensystems erreicht, so kann das Objekt mit `glSphere()` erstellt werden. Die Beschreibung der Modellierung der Erde im folgenden Abschnitt enthält ein Beispiel hierzu.

Um nun ein weiteres Objekt an einer anderen Stelle im Raum zu positionieren, empfiehlt es sich, das ursprüngliche Koordinatensystem mit `LoadIdentity()` wiederherzustellen und wiederum durch entsprechende Verschiebung bzw. Rotation die Positionierung durchzuführen.

Um die Modellierung zu vereinfachen, wird festgelegt, dass die Blickrichtung des initialisierten OpenGL Fensters der Frühlingspunkt ist und die Y-Achse zum Himmelsnordpol weist. Die Erde wird im Koordinatenursprung positioniert, sodass das Koordinatensystem dem Q-System mit vertauschten Achsen entspricht. Die Transformationsmatrix vom Q- in das G-System lautet:

$$(\_)_{GQ} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \end{pmatrix} \quad (5.1)$$

#### 5.1.2.1 Erde

Um die Entstehung der Simulationsumgebung deutlicher zu zeigen, wurde die Kameraperspektive verändert. Die Kamera wurde 60.000 km vom Ursprung des G-Koordinatensystems in  $+Z_{(G)}$  -Richtung bewegt, Blickrichtung ist weiterhin der Frühlingspunkt.

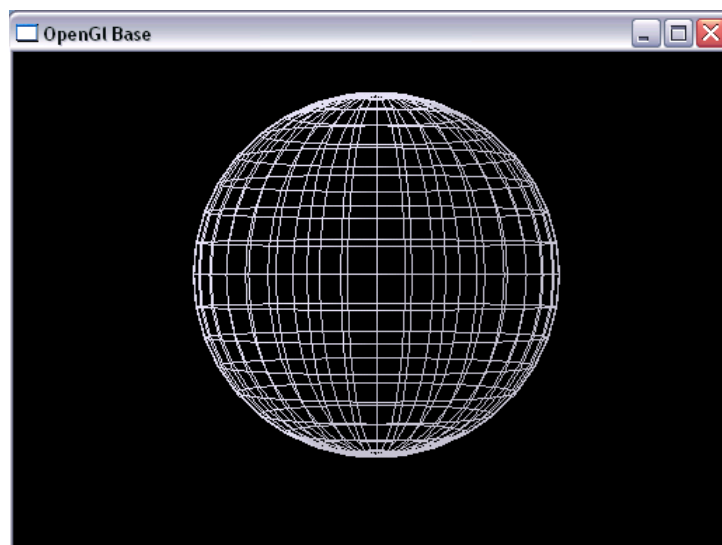


Abbildung 5-2: Erdkugel als Gitternetz

Die Modellierung der Erde beginnt mit der Rotation von  $90^\circ$  des G-Koordinatensystems um die  $X_{(G)}$ -Achse. Die Z-Achse des daraus resultierenden Koordinatensystems weist damit nach oben. Dadurch ist wiederum gewährleistet, dass die Pole der entstehenden Kugel wie gefordert auf der  $Z_{(Q)}$ -Achse liegen. Eine Ver-

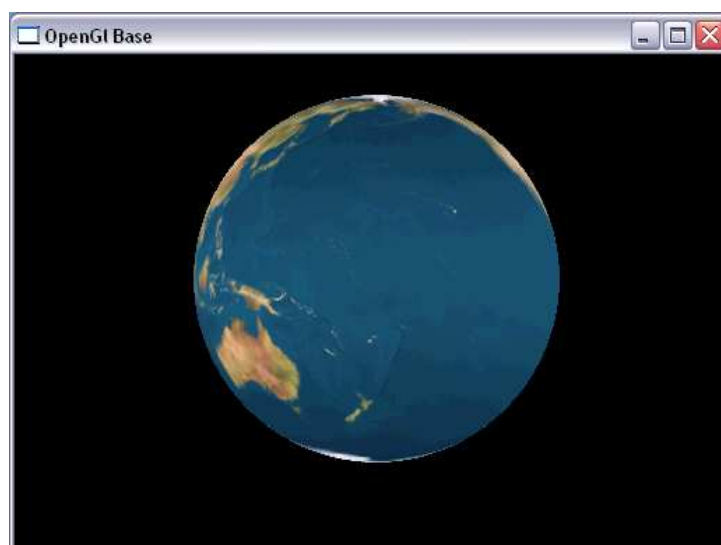


**Abbildung 5-3:** Zweidimensionales Bild der Erdoberfläche

schiebung des Koordinatensystems mit `glTranslate()` ist nicht notwendig, da der Erdmittelpunkt ja im Ursprung des Q-Koordinatensystems liegen soll. Anschließend kann die Erdkugel mit `glSphere()` erstellt werden, wobei als Radius der mittlere Erdradius von 6378km angegeben wird. Für die Anzahlen der Längen- und Breitenunterteilungen werden Werte von 72 bzw. 144 festgelegt, sodass die Abstände der Breitenkreise und die Abstände der Längengrade am Äquator  $2,5^\circ$  betragen.

Abbildung 5-2 zeigt beispielhaft die Erdkugel als Gitternetz. Die Anzahl der Längen- und Breitenlinien wurde verringert, damit die Gitterstruktur erkennbar bleibt.

Ein Erdmodell in Form eines Gitters ist zur Unterstützung der Satellitenmanöver jedoch nicht geeignet. Für den Operator sollen auf dem simulierten Kamerabild später



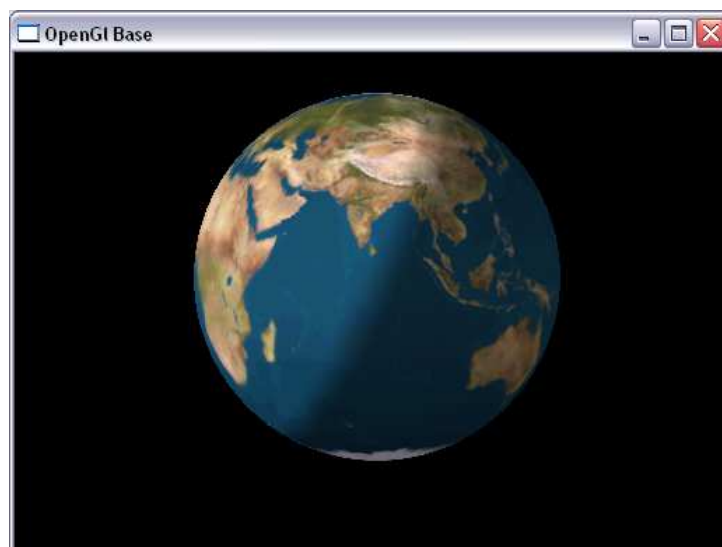
**Abbildung 5-4:** Erdkugel mit Textur

Küstenlinien und landschaftliche Strukturen erkennbar sein, um ihm die Orientierung zu ermöglichen. Zur Lösung dieses Problems kann eine weitere OpenGL-Funktion eingesetzt werden:

OpenGL bietet die Möglichkeit, sogenannte „Texturen“ auf die Oberfläche von 3D-Objekten zu legen. Eine Textur entsteht, indem ein zweidimensionales Bild, welches beispielsweise in Form einer Bitmap vorliegt, der gekrümmten Körperoberfläche angepasst wird. Abbildung 5-3 zeigt ein solches zweidimensionales Bild der Erdoberfläche. Nach dem Aufbringen dieses Bildes auf die Gitterkugel ergibt sich die Erdkugel wie in Abbildung 5-4 dargestellt. Im letzten Schritt wird die Erdkugel schließlich um ihre Rotationsachse richtig orientiert, d.h. sie wird derart rotiert, dass der Winkel zwischen der Nullmeridianebene und der Richtung zum Frühlingspunkt in der Simulation dem zum Simulationszeitpunkt aktuellen Greenwicher Stundenwinkel ( $\theta_g$ ) entspricht. Die Berechnung von  $\theta_g$  erfolgt unter Einsatz der SGP-4-Bibliothek von Kelso [13], welche als Quellcode im Internet frei verfügbar ist. Es handelt sich dabei um eine umfangreiche Sammlung von mathematischen Routinen, welche im Zusammenhang mit der Berechnung von Satellitenbahnen von Interesse sind. Dieser Weg wurde gewählt, um den Programmieraufwand für die Simulationssoftware zu begrenzen. SGP-4 hat sich als Standard für Bahnberechnungen auf Basis der sogenannten Norad-Two-Line-Elements (TLE) etabliert und findet u.a. Verwendung in Bahnrechnungsprogrammen wie Traksat oder STS-Plus.

#### 5.1.2.2 Sonne

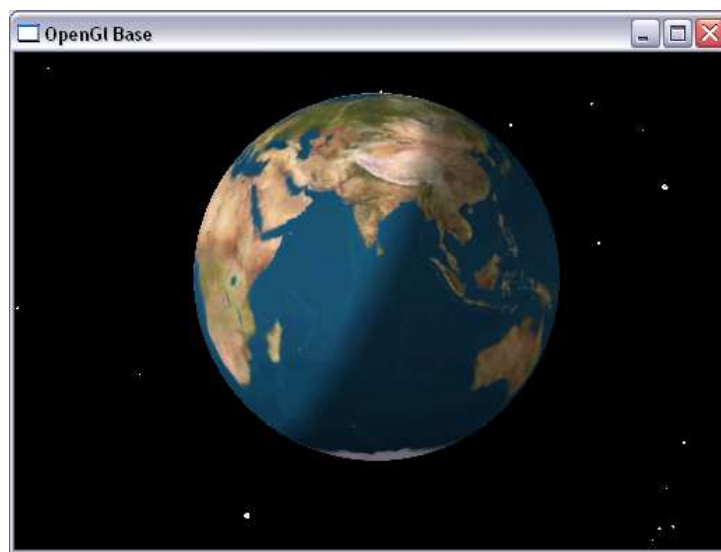
Zur Modellierung der Sonne wird zunächst der Positionsvektor der Sonne im Q-System für den Simulationszeitpunkt berechnet. Auch hierfür stellt die SGP-4-Bibliothek eine fertige Routine zur Verfügung. Der Positionsvektor wird dann noch



**Abbildung 5-5:** Von der Sonne beschienene Erdkugel

um einen Faktor 100 gekürzt, da die Entfernung von Erde und Sonne in der Simulation nicht notwendigerweise der echten Entfernung entsprechen muss. Wichtig ist lediglich, dass die Entfernung groß ist gegenüber dem Erddurchmesser, sodass das Sonnenlicht im erdnahen Bereich als parallel erscheint. An dieser Position wird die Sonne anschließend, wiederum mithilfe des Befehls `GluSphere()`, als weiße Kugel erstellt.

Für die Simulation des Sonnenlichts kommt eine weitere Funktion von OpenGL zum Einsatz: Mit OpenGL ist es möglich, Lichtquellen im virtuellen Raum zu positionieren. Hierfür genügt eine Koordinatenangabe und eine Festlegung der Intensität der Lichtquelle. Um das Sonnenlicht zu simulieren, wird eine solche Lichtquelle an die



**Abbildung 5-6:** Simulation mit Erde, Sonne und Sternen

Sonnenposition gesetzt. Nun sind auf der Erdkugel Tag- und Nachtseite voneinander zu unterscheiden (siehe Abbildung 5-5). Zusätzlich zum Positionieren von Lichtquellen bietet OpenGL auch noch die Möglichkeit, eine Hintergrundbeleuchtung einzustellen. Das Niveau der Hintergrundbeleuchtung wird so gewählt, dass der Operator auch auf der Nachtseite der Erde Küstenlinien und andere Strukturen erkennen kann.

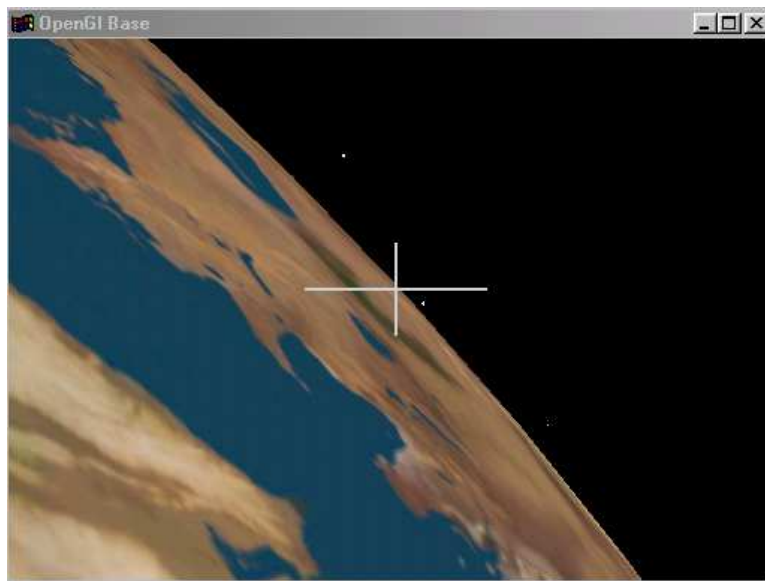
#### 5.1.2.3 Sterne

Für die Darstellung des Sternenhimmels in der Simulation werden die 2500 hellsten Sterne genau wie die Sonne als weiße Kugeln erzeugt, wobei sich der Radius der Kugel an der Magnitude des jeweiligen Sterns orientiert. Ausgangspunkt sind die Sternkoordinaten aus dem Sternkatalog. Jeder Stern wird nun in Verlängerung seines Richtungsvektors in einer Entfernung von  $10^7$  km vom Erdmittelpunkt positioniert. Diese Entfernung ist wiederum beliebig gewählt und soll nur sicherstellen, dass das Sternlicht in Erdnähe parallel einfällt. Sämtliche simulierte Sterne befinden sich somit

auf einer Sphäre mit einem Radius von  $10^7$  km in deren Mittelpunkt sich die Erde befindet. Abbildung 5-6 zeigt die Simulationsumgebung mit Erde Sonne und Sternenhimmel.

#### 5.1.2.4 Satellit

Da die Simulation zu jeder Zeit die Perspektive der Nutzlastkamera bzw. des Sternsensors an Bord des Satelliten darstellen soll, muss nun noch die „Kamera“ im simulierten Raum positioniert und orientiert werden. Die Position des Satelliten bzw. der Betrachterkamera wird ebenfalls mit Routinen der SGP-4-Bibliothek bestimmt. Die Positionierung erfolgt wiederum mit dem Befehl `glTranslate()`. Die Satellitenlage



**Abbildung 5-7:** Simulierter Blick aus dem Satelliten

wird zu Beginn der Simulation auf den Frühlingspunkt ausgerichtet, d.h. Rektaszension und Deklination werden auf 0 gesetzt. Der Winkel um die Bildachse (Azimut) wird ebenfalls auf 0 gesetzt. Es ergibt sich also der Blick aus dem Satelliten auf den Frühlingspunkt, wobei sich der Himmelsnordpol über dem Betrachter befindet. Während der Simulation kann die Satellitenlage dann beliebig verändert werden. Nach der erfolgreichen Lageermittlung des echten Satelliten mittels Sternsensor, kann diese Ist-Lage von der Simulation übernommen werden und alle folgenden Schwenkmanöver können parallel zum Überflug in der Simulation mitgekoppelt werden. Abbildung 5-7 zeigt beispielhaft den simulierten Blick aus dem Satelliten während eines Überflugs.

Die Veränderung der Satellitenlage (bzw. der Kamerablickrichtung) erfolgt, indem vor dem Zeichnen der Simulationsszene (Sonne, Sterne und Erde) das OpenGL-Koordinatensystem rotiert und verschoben wird. Die Rotationen und Verschiebungen ergeben sich direkt aus der aktuellen Lage und Position des Satelliten. Soll die Kamera von der Position  $(x,y,z)_q$  mit der Lage  $(RA,DE,AZ)$  die Szene betrachten, so



wird das G-System zunächst in der angegebenen Reihenfolge um -Az, -De und -Ra rotiert und um  $(-x, -y, -z)_q$  verschoben, bevor die Objekte gezeichnet werden.

Damit die so erstellte Umgebung zu einer Echtzeitsimulation wird, durchläuft der Rechner eine Schleife, in der zuerst die für die Darstellung relevanten Parameter (Satellitenposition, Sonnenposition, Greenwicher Stundenwinkel) zur aktuellen Systemzeit berechnet werden und dann die OpenGL-Szene entsprechend der obigen Beschreibung neu gezeichnet wird. Auf diese Weise wird die OpenGL-Szene mehrmals pro Sekunde aktualisiert und erscheint daher animiert.

### 5.1.3 Lagebeschreibung

Für die Lagebeschreibung des Satelliten bzw. der Kamera wurden Sternwinkel (RA, De und AZ) gewählt, da sie eine besonders anschauliche Beschreibung ermöglichen. Es war dadurch auch möglich, während der Entwicklungsphase Fehler in der Simulation aufzudecken, da bei einem Schwenk durch den Sternenhimmel Rektaszension und Deklination der im Bild befindlichen Sterne mit der berechneten Blickrichtung der Kamera abgeglichen werden konnten.

Für die Berechnung der Schwenkmanöver und für die flüssige Animation derselben wurde jedoch auf die sogenannten Euler-Parameter oder auch Quaternionen zurückgegriffen, welche mathematisch bequemer zu handhaben sind. Es werden in der Software also parallel zwei verschiedene Lagerepräsentationen verwendet: Einerseits die Sternwinkel an den Schnittstellen zum Benutzer und zum Sternsensor und andererseits Quaternionen für die Simulationsberechnungen im Hintergrund. Im folgenden Abschnitt soll dieses Verfahren genauer erläutert werden.

Da die Simulation dem Zweck dient einen echten Überflug zu unterstützen und die echte Satellitenlage mitzukoppeln, muss zu Beginn des Überflugs die durch den Sternsensor ermittelte Lage, welche von der Software MusterPr im Format von Sternwinkeln ausgegeben wird, von der Simulation übernommen werden. Hierfür ist eine Umrechnung der Sternwinkel in einen Ist-Lage-Quaternion notwendig. Für die Umrechnung werden Formeln aus [14] herangezogen. In ihnen tauchen jedoch keine Sternwinkel auf, sondern nur Eulerwinkel. Der einzige Unterschied ist, dass das Vorzeichen der Deklination umgekehrt definiert ist als bei dem entsprechenden Euler Winkel  $\vartheta$ : Die mathematisch positive Drehrichtung um die Y-Achse im Q-System wäre eine Nickbewegung nach unten. Die Deklination ist aber per Definition auf der nördlichen Himmelshalbkugel positiv. Es gilt:

$$RA = \varphi \quad (5.2)$$

$$DE = -\vartheta \quad (5.3)$$

$$AZ = \psi \quad (5.4)$$

Nun kann die Berechnung des Ist-Lage-Quaternions erfolgen. Die Gleichungen für seine Komponenten lauten:

$$q_1 = \cos\left(\frac{\varphi}{2}\right) \cdot \cos\left(\frac{\vartheta}{2}\right) \cdot \sin\left(\frac{\psi}{2}\right) - \sin\left(\frac{\varphi}{2}\right) \cdot \sin\left(\frac{\vartheta}{2}\right) \cdot \cos\left(\frac{\psi}{2}\right) \quad (5.5)$$

$$q_2 = \cos\left(\frac{\varphi}{2}\right) \cdot \sin\left(\frac{\vartheta}{2}\right) \cdot \cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\varphi}{2}\right) \cdot \cos\left(\frac{\vartheta}{2}\right) \cdot \sin\left(\frac{\psi}{2}\right) \quad (5.6)$$

$$q_3 = \sin\left(\frac{\varphi}{2}\right) \cdot \cos\left(\frac{\vartheta}{2}\right) \cdot \cos\left(\frac{\psi}{2}\right) - \cos\left(\frac{\varphi}{2}\right) \cdot \sin\left(\frac{\vartheta}{2}\right) \cdot \sin\left(\frac{\psi}{2}\right) \quad (5.7)$$

$$q_4 = \cos\left(\frac{\varphi}{2}\right) \cdot \cos\left(\frac{\vartheta}{2}\right) \cdot \cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\varphi}{2}\right) \cdot \sin\left(\frac{\vartheta}{2}\right) \cdot \sin\left(\frac{\psi}{2}\right) \quad (5.8)$$

Soll nun der Satellit ausgehend von dieser Ist-Lage ein Rotationsmanöver um eine seiner Achsen ausführen, so wird dieses Manöver von der Simulationssoftware ebenfalls in Quaternionenform gebracht. Ein Schwenk um einen Winkel  $\alpha$  bezüglich einer Achse bzw. eines Einheitsvektors  $\underline{e}$  mit den Komponenten  $e_x, e_y$ , und  $e_z$  lässt sich folgendermaßen als Quaternion darstellen:

$$q_1 = e_x \cdot \sin\left(\frac{\alpha}{2}\right) \quad (5.9)$$

$$q_2 = e_y \cdot \sin\left(\frac{\alpha}{2}\right) \quad (5.10)$$

$$q_3 = e_z \cdot \sin\left(\frac{\alpha}{2}\right) \quad (5.11)$$

$$q_4 = \cos\left(\frac{\alpha}{2}\right) \quad (5.12)$$

Bezogen auf das Satellitenmanöver stellt  $\underline{e}$  die Rotationsachse des Satelliten dar, also je nach Manöver seine X,Y oder Z-Achse.

Die sich durch dieses Manöver ergebende neue Lage bzw. den resultierende Lagequaternion  $q_C$  kann man aus dem ursprünglichen Lagequaternion  $q_A$  und dem Manöverquaternion  $q_B$  nun nach folgender Formel bestimmen:

$$q_C = \begin{pmatrix} q_{A1} \cdot q_{B4} + q_{A2} \cdot q_{B3} - q_{A3} \cdot q_{B2} + q_{A4} \cdot q_{B1} \\ -q_{A1} \cdot q_{B3} + q_{A2} \cdot q_{B4} + q_{A3} \cdot q_{B1} + q_{A4} \cdot q_{B2} \\ q_{A1} \cdot q_{B2} - q_{A2} \cdot q_{B1} + q_{A3} \cdot q_{B4} + q_{A4} \cdot q_{B3} \\ -q_{A1} \cdot q_{B1} - q_{A2} \cdot q_{B2} - q_{A3} \cdot q_{B3} + q_{A4} \cdot q_{B4} \end{pmatrix} \quad (5.13)$$

Auf diese Weise können beliebig viele Manöver aneinander gereiht werden. Es müssen jedoch nach jedem Schwenk des Satelliten auch seine Achsen entsprechend transformiert werden. Dafür wird folgende Gleichung herangezogen:

$$\begin{bmatrix} v_x' \\ v_y' \\ v_z' \end{bmatrix} = (2q_4^2 - 1) \cdot \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} + 2 \cdot (q_1 v_x + q_2 v_y + q_3 v_z) \cdot \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} + 2q_4 \cdot \begin{bmatrix} q_2 v_z - q_3 v_y \\ q_3 v_x - q_1 v_z \\ q_1 v_y - q_2 v_x \end{bmatrix} \quad (5.14)$$

Die Gleichung stellt den allgemeinen Fall der Transformation eines Vektors  $\underline{v}$  in einen Vektor  $\underline{v}'$  mittels eines Quaternions  $\underline{q}$  dar. Im vorliegenden Fall steht  $\underline{v}$  für eine Achse des zum Frühlingspunkt ausgerichteten Satelliten (mit RA=DE=AZ=0). Der Quaternion  $\underline{q}$  beschreibt die aktuelle Satellitenlage.  $\underline{v}'$  steht schließlich für die sich ergebende transformierte Satellitenachse.

Nach der Berechnung des neuen Lagequaternions muss nun noch die Umrechnung in Sternwinkel erfolgen, damit diese dem Benutzer der Software angezeigt werden können und damit die Simulationsszene neu gezeichnet werden kann.

Im ersten Schritt wird die zum aktuellen Lagequaternion gehörige Rotationsmatrix gebildet:

$$(\_)_{SQ} = \begin{bmatrix} 2(q_1 q_3 + q_2 q_4) & 2(q_2 q_3 - q_1 q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \\ -2(q_1 q_2 - q_3 q_4) & q_1^2 - q_2^2 + q_3^2 - q_4^2 & -2(q_2 q_3 + q_1 q_4) \\ q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1 q_2 + q_3 q_4) & 2(q_1 q_3 - q_2 q_4) \end{bmatrix} \quad (5.15)$$

Durch den Vergleich dieser Matrix mit der aus Gleichung 4.1 ergeben sich folgende Bestimmungsgleichungen für die Sternwinkel:

$$DE = \arcsin(2(q_1 q_3 - q_2 q_4)) \quad (5.16)$$

$$AZ = \arctan 2 \left( 2(q_2 q_3 + q_1 q_4), -q_1^2 - q_2^2 + q_3^2 + q_4^2 \right) \quad (5.17)$$

$$RA = \arctan 2 \left( 2(q_1 q_2 + q_3 q_4), q_1^2 - q_2^2 - q_3^2 + q_4^2 \right) \quad (5.18)$$

## 5.2 Manöverberechnung

### 5.2.1 Berechnung der Schwenkwinkel

In diesem Unterkapitel sollen die Berechnungen der Schwenkmanöver vorgestellt werden, welche notwendig sind, um den Satelliten vorzustabilisieren, d.h. grob auf ein Ziel auf der Erde auszurichten.

Zunächst muss für die folgenden Berechnungen noch ein erdfestes Koordinatensystem eingeführt werden, mit dem die Position eines Ziels auf der Erdoberfläche beschrieben werden kann, das sogenannte E-System [11]. Der Koordinatenursprung dieses Systems liegt im Erdmittelpunkt, die X-Achse weist zum Schnittpunkt von Nullmeridian und Äquator und die Z-Achse ist zum Nordpol ausgerichtet. Die Y-Achse ergänzt das System zum Rechtssystem.

Ausgangspunkt für die Berechnung ist die Eingabe von geographischer Länge und Breite des Ziels, sowie die Festlegung des genauen Zeitpunktes zu dem der Satellit auf das Ziel ausgerichtet sein soll. Der Ortsvektor  $\underline{B}$  des Ziels, mit der Länge  $\lambda$  und Breite  $\beta$ , lässt sich im E-Koordinatensystem darstellen als

$$\underline{B}_E = r \cdot \begin{pmatrix} \cos \lambda \cdot \cos \beta \\ \sin \lambda \cdot \cos \beta \\ \sin \beta \end{pmatrix}, \quad (5.19)$$

wobei  $r$  den Erdradius am Zielort meint. Zur Vereinfachung wird für die Berechnung dieses Vektors stets der mittlere Erdradius von 6378 km angenommen.

Um  $\underline{B}$  nun vom E-System in das inertial-feste Q-System zu überführen, wird folgende Umrechnungsmatrix [11] verwendet:

$$\left( \underline{\quad} \right)_{QE} = \begin{pmatrix} \cos \theta_g & -\sin \theta_g & 0 \\ \sin \theta_g & \cos \theta_g & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (5.20)$$

Der Greenwicher Stundenwinkel  $\theta_g$  wird dabei wieder mit Hilfe der entsprechenden Funktion aus der SGP-4-Bibliothek für den Ausrichtungszeitpunkt bestimmt. Der Ortsvektor des Ziels lautet damit im Q-System

$$\underline{B}_Q = r \cdot \begin{pmatrix} \cos \beta \cdot (\cos \theta_g \cdot \cos \lambda - \sin \theta_g \cdot \sin \lambda) \\ \cos \beta \cdot (\sin \theta_g \cdot \cos \lambda + \cos \theta_g \cdot \sin \lambda) \\ \sin \beta \end{pmatrix}. \quad (5.21)$$

Aus  $\underline{B}$  und dem Ortsvektor des Satelliten  $\underline{r}$  (zum Aufnahmezeitpunkt), dessen Bestimmung wiederum unter Heranziehung der SGP-4-Bibliothek erfolgt, ergibt sich der Vektor  $\underline{\rho}$ , der vom Satelliten zum Ziel auf der Erdoberfläche weist, als

$$\underline{\rho}_Q = \underline{B}_Q - \underline{r}_Q. \quad (5.22)$$

Abbildung 5-8 veranschaulicht diese Beziehung.



Zur Bestimmung der Schwenkwinkel  $MX$  und  $MY$  wird der Vektor  $\underline{\rho}$  zunächst mit Hilfe der SQ-Transformationsmatrix aus Gleichung 4.1 ins S-System überführt:

$$\underline{\rho}_s = \underline{(\ )}_{SQ} \cdot \underline{\rho}_Q \quad (5.23)$$

Durch Normierung von  $\underline{\rho}_s$  erhält man dessen Einheitsvektor  $\underline{u\rho}_s$ . Dieser wiederum lässt sich nun in Kugelkoordinaten darstellen als

$$\underline{u\rho}_s = \begin{pmatrix} Xu \\ Yu \\ Zu \end{pmatrix} = \begin{pmatrix} \sin MY \\ -\sin MX \cos MY \\ \cos MX \cos MY \end{pmatrix}. \quad (5.24)$$

Damit ergeben sich die Schwenkwinkel schließlich aus seinen Komponenten zu:

$$MY = \arcsin(Xu) \quad (5.25)$$

$$MX = \arctan 2(-Yu, Zu) \quad (5.26)$$

Nach Ausführung dieser Schwenks ist die optische Achse nun entlang des Zielvektors ausgerichtet, die Lage des vorstabilisierten Satelliten um die Bildachse ist beliebig. Letzteres ist, wie in Kapitel 2.1 bereits angedeutet, für die interaktive Lageregelung eigentlich kein Problem. Es hat sich in der Praxis jedoch herausgestellt, dass ein eingenordetes Bild der Nutzlastkamera die Orientierung während des Überflugs erleichtert. Aus diesem Grund wird noch ein drittes Teilmanöver durchgeführt, für das allerdings kein zusätzlicher Rechenaufwand anfällt. Es wird lediglich der Azimut auf  $0^\circ$  gebracht. Hier wird ebenfalls der kürzeste Schwenkweg gewählt.

Für  $0^\circ < AZ \leq 180^\circ$ :

$$\alpha_{M3} = -AZ \quad (5.27)$$

Für  $180^\circ < AZ \leq 360^\circ$

$$\alpha_{M3} = 360^\circ - AZ \quad (5.28)$$

Dies sorgt bei steilen Überflügen dafür, dass das Bild ausreichend gut eingenordet ist, die Landschaft also von oben nach unten durch das Bild läuft.

Neben der Vorstabilisierung auf ein Ziel auf der Erdoberfläche hat der Benutzer mit dieser Software auch die Möglichkeit, mit dem Satelliten einen bestimmten Punkt am Sternenhimmel anzupeilen. Diese Funktionalität wurde implementiert, um den in Kapitel 8.1.2 dokumentierten Vorversuch durchführen zu können. Mithilfe dieses Vorversuchs wurde die Genauigkeit des Vorstabilisierungsverfahrens vor der Durch-

führung der Versuche zur interaktiven Lageregelung überprüft. Der einzige Unterschied zu der beschriebenen Vorstabilisierung ist, dass der Zielvektor  $\underline{p}$  anders bestimmt wird. Bei Angabe der Ziel-Sternkoordinaten in RA und DE ergibt sich  $\underline{p}$  direkt aus Gleichung 4.2. Die Vorgabe eines Beobachtungszeitpunktes fällt weg, da  $\underline{p}$  sich zeitlich nicht verändert. Der weitere Rechenweg entspricht dem oben beschriebenen Ansatz.

## 5.2.2 Benutzerschnittstelle

Abbildung 5-10 zeigt die Benutzerschnittstelle für die Manöversoftware. Oben links im Fenster (unter a)) wird die aktuelle Satellitenlage in Sternwinkeln angezeigt. Die Anzeige wird während der Manöver in Echtzeit aktualisiert.

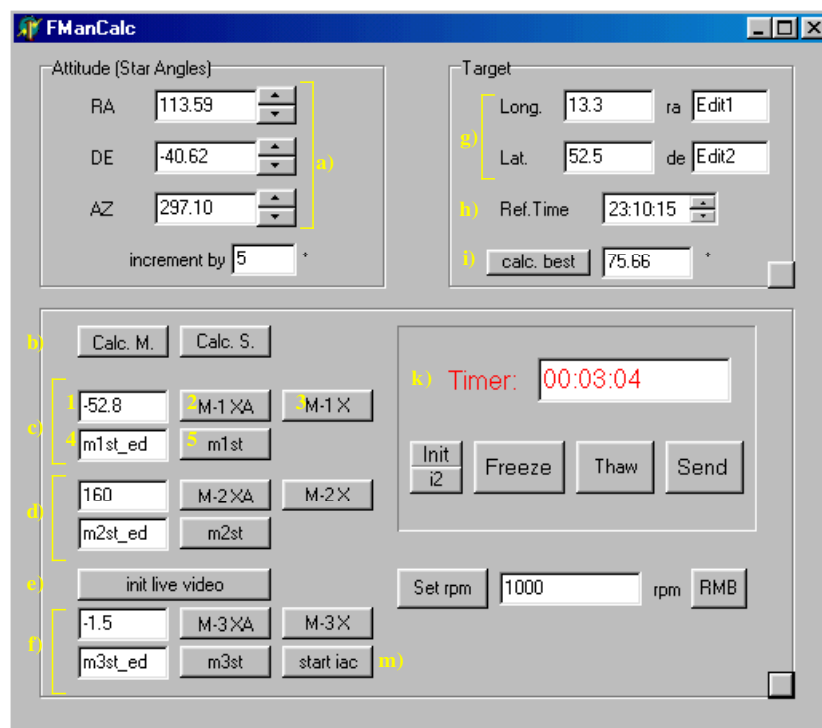


Abbildung 5-10: Benutzeroberfläche der Manöversoftware

Die Angabe des Ziels erfolgt über die Eingabefelder g). Auf der linken Seite können Länge und Breite eines Zielorts auf der Erde angegeben werden, alternativ dazu können rechts Rektaszension und Deklination eines Ziels im Sternenhimmel angegeben werden.

Soll ein Ziel auf der Erde aufgenommen werden, so muss unter h) der Zeitpunkt festgelegt werden, für den die Berechnung der Manöver durchgeführt werden soll. Die Zeit kann der Operator entweder direkt vorgegeben, oder er kann die Schaltfläche i) anwählen. Letzteres führt dazu, dass die Software automatisch den Zeitpunkt bestimmt, zu dem der Satellit bezogen auf den Zielort die höchste Elevation bzw. die geringste Schrägentfernung hat. Für die Berechnung des Zeitpunktes wird eine

Schleife durchlaufen, in der zunächst die Position des Satelliten in Sekundenschritten, beginnend bei der aktuellen Uhrzeit, berechnet wird. Die Position wird anschließend in das Horizontsystem des Zielorts transformiert. Die Austrittsbedingung aus der Schleife ist erfüllt, sobald ein relatives Maximum der Elevation festgestellt wird. Dieser günstigste Aufnahmezeitpunkt wird dann im Feld h) angezeigt.

Durch Anklicken einer der beiden Schaltflächen bei b) wird die Manöverberechnung gestartet. Die linke Fläche löst die Berechnung für Ziele auf der Erdoberfläche aus, die Rechte für den Sternhimmel.

Gleichzeitig wird der Countdown Timer k) gestartet. Er zeigt die noch verbleibende Zeit bis zum über h) festgelegten Aufnahmezeitpunkt an.

Die berechneten Winkel für die drei Teilmanöver erscheinen in den Bereichen c), d) und f) jeweils oben links im Feld 1.

Der Druck auf den Knopf 2 rechts daneben bewirkt, dass das zugehörige Winkelkommando einerseits zum realen Satelliten gefunkt wird und andererseits in der Simulation mitgekoppelt wird.

Falls der Befehl den Satelliten, z.B. wegen einer gestörten Funkverbindung, nicht erreicht, besteht die Möglichkeit, den Befehl zu wiederholen, ohne dass die Simulation fälschlicherweise erneut mitkoppelt. Hierfür ist Knopf 3 hinzugefügt worden.

Knopf 5 bewirkt die Abfrage der jeweiligen ACS-Einheit nach dem tatsächlich bereits ausgeführten Winkel. Das Ergebnis der Abfrage wird im Telemetriefeld 4 ausgegeben. Mithilfe dieser Abfragemöglichkeit lässt sich sicherstellen, dass alle Winkel korrekt und vollständig ausgeführt werden.

Mit der Schaltfläche e) wird die Nutzlastkamera für die Live-Video-Modus initialisiert. Zusätzlich wird auch die Bildübertragung via S-Band aktiviert. Sie wird sinnvollerweise erst nach dem zweiten Schwenkmanöver betätigt, da sich die Bodenstation in der Regel erst nach Abschluss dieses Manövers innerhalb der Keule der S-Band-Antenne des Satelliten befindet. Die Schaltfläche ist daher unterhalb des Schaltfeldes zum zweiten Manöver angeordnet.

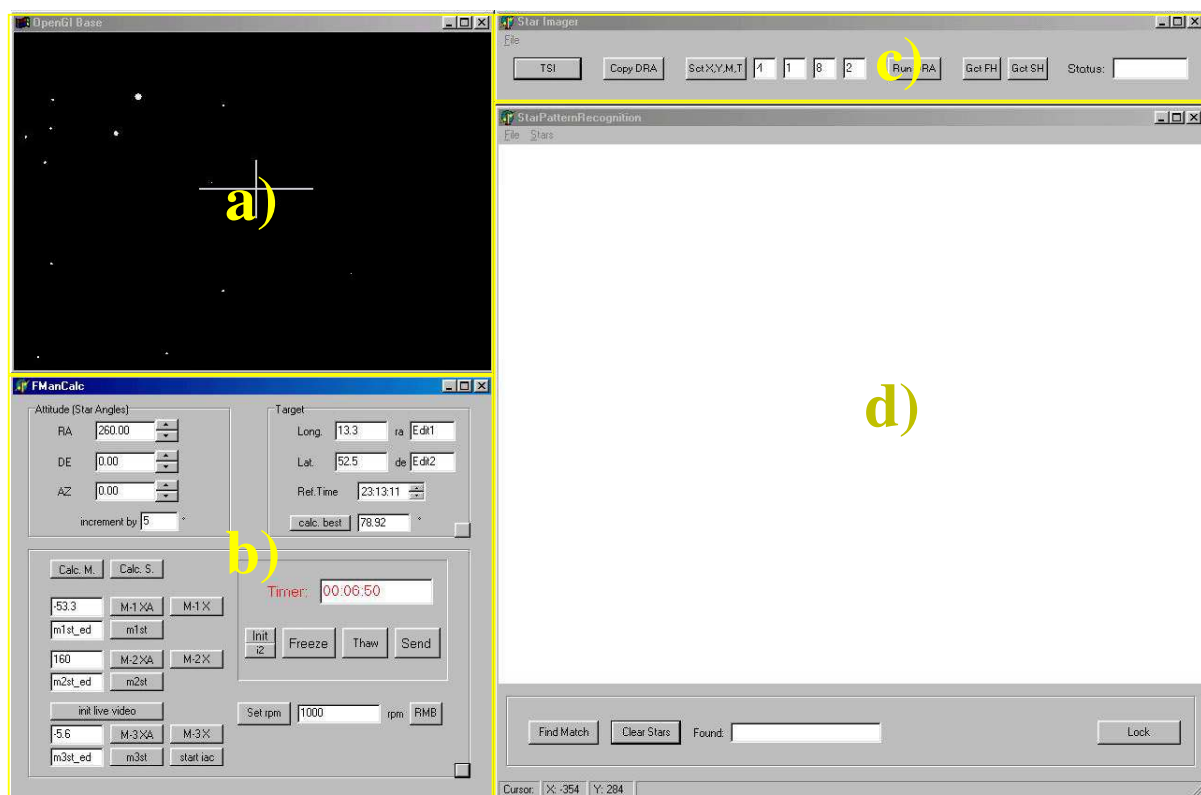
Schaltfläche m) leitet den Übergang von der Vorstabilisierung zu der interaktiven Lageregelung ein. Sie startet die Software zur interaktiven Lageregelung, deren Beschreibung in Kapitel 7 folgt.



## 6 Integrierte Vorstabilisierungssoftware

### 6.1 Benutzerinterface

Durch Integration der in den Kapiteln 3 bis 5 vorgestellten Softwarekomponenten ergibt sich ein Programmpaket, welches es ermöglicht, den Satelliten während eines Nachtüberflugs, ausgehend von einer beliebigen unbekannten Lage, der Lost-In-Space-Konfiguration, auf ein Ziel vorauszurichten.



**Abbildung 6-1:** Benutzeroberfläche der integrierten Vorstabilisierungssoftware

Die Bedienoberfläche dieser Vorstabilisierungssoftware ist in Abbildung 6-1 dargestellt und setzt sich aus den in den vorangegangenen Kapiteln bereits beschriebenen Oberflächen zusammen. Oben rechts angeordnet ist das Fenster c) zum Bedienen des Sternsensors, welches in Kapitel 3 ausführlich beschrieben wurde. Im Zuge der Integration zur Vorstabilisierungssoftware wurde hier eine Schaltfläche hinzugefügt, die das Programm PHTSI zur Sternbildaufnahme auslöst (siehe Kapitel 3.2.1). Dadurch wird der Satellit zuerst inertial stabilisiert und anschließend wird ein Sternbild aufgenommen. Dieses im Bildspeicher der Kamera befindliche Sternbild wird dann in der in Kapitel 3 beschriebenen Weise reduziert und zur Bodenstation übertragen.

Eine weitere Änderung des Fensters c) besteht darin, dass die Anzeige des reduzierten Sternbildes in das Fenster d) ausgelagert wurde. Das Sternbild wird also direkt an die Mustererkennungssoftware übergeben. Der Operator kann dann in Fenster d), wie in Kapitel 4 beschrieben, die Sterne markieren und die Auswertung bzw. Lageermittlung vornehmen. Ist die Lage des Satelliten ermittelt, so wird sie wiederum auf Knopfdruck des Operators von der Simulationssoftware übernommen. Der virtuelle Satellit übernimmt damit die Lage des realen Satelliten. Im OpenGL-Fenster a), was zu jeder Zeit das Blickfeld des simulierten Satelliten darstellt (siehe Kapitel 5), erscheint daraufhin derselbe Ausschnitt des Sternenhimmels wie in Fenster d). Zusätzlich wird die Satellitenlage auch in Fenster b) angezeigt. Mithilfe der Eingabefelder und Schaltflächen von Fenster b) werden schließlich, wie unter Kapitel 5 beschrieben, die notwendigen Schwenks für die Vorstabilisierung des Satelliten berechnet und ausgeführt.

## 6.2 Ablauf der Vorstabilisierung

In diesem Unterkapitel soll die Funktionsweise der integrierten Vorstabilisierungssoftware bzw. das Zusammenspiel der einzelnen Komponenten anhand des Überflugs vom 27.02.2003 um 23:05 UTC näher erläutert werden. Es sei an dieser Stelle darauf hingewiesen, dass die unter 5.2.1 beschriebenen Schwenkreihenfolge „X-Y“ erst später implementiert wurde. Zum Zeitpunkt des im Folgenden beschriebenen Überflugs wurde zunächst um Z- und anschließend um X geschwenkt.

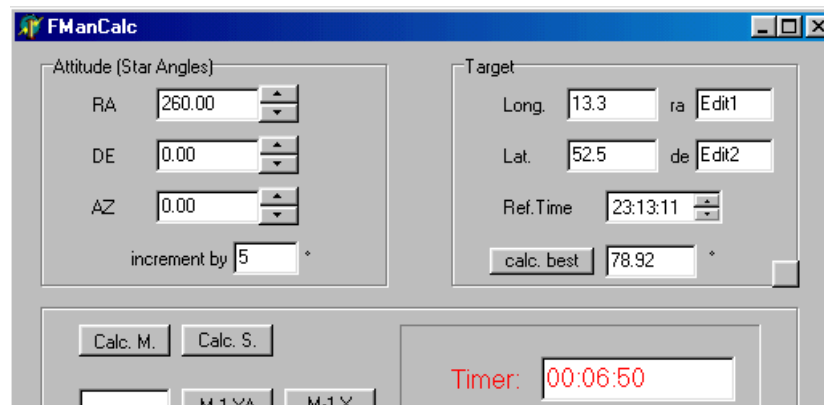
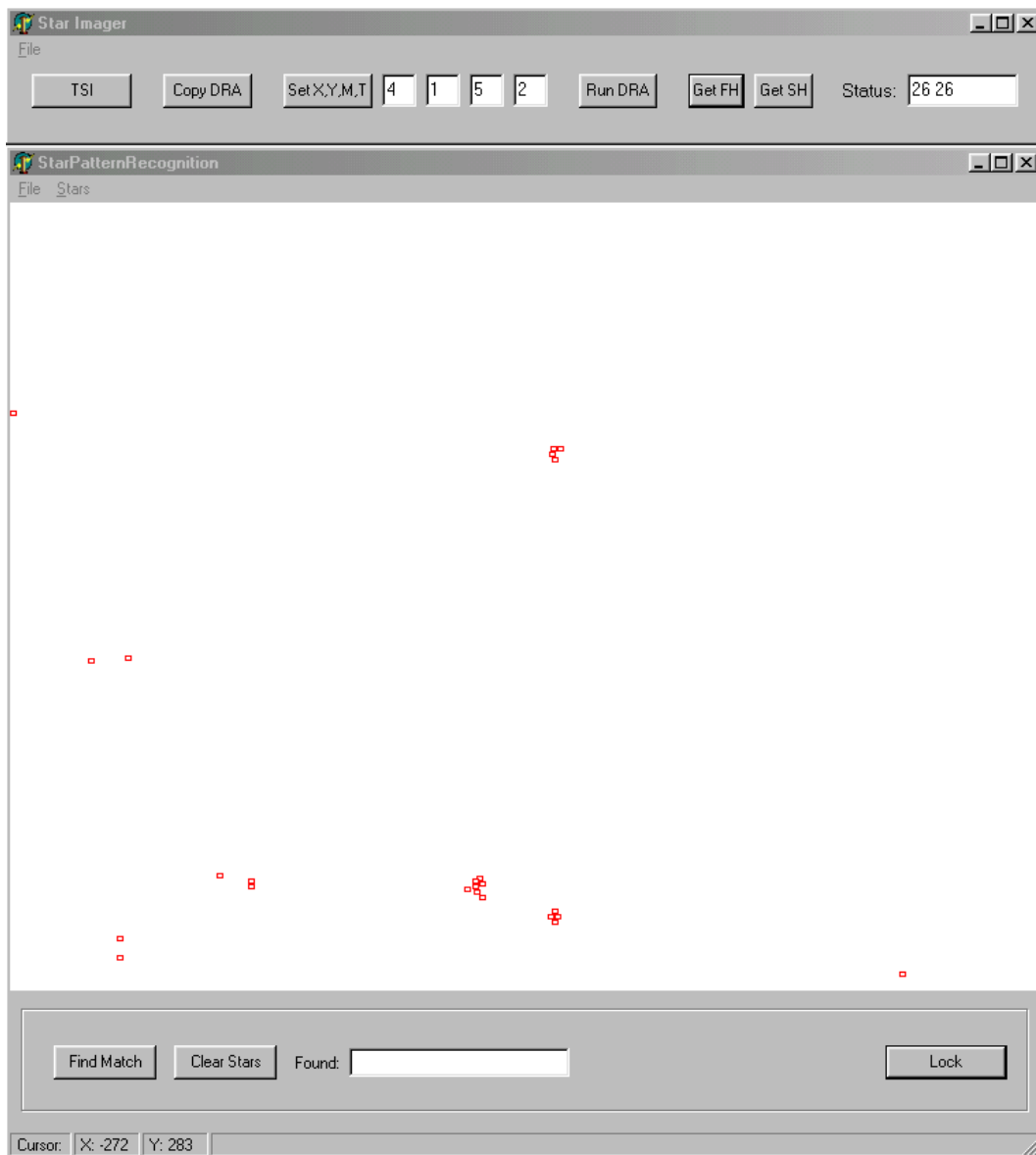


Abbildung 6-2: Fenster b), optimaler Aufnahmezeitpunkt

Zur Verbesserung der Übersicht wird in jedem Abschnitt der Beschreibung jeweils nur der relevante Teil der Benutzeroberfläche abgebildet, der Operator hat während des Überflugs jedoch zu jeder Zeit alle vier Fenster, wie in Abbildung 6-1 gezeigt, vor sich.

Noch bevor der Operator Funkkontakt zum Satelliten hat, werden die Koordinaten des Ziels eingegeben. In diesem Fall soll Berlin mit  $\lambda=13,3^\circ$  und  $\beta=52.5^\circ$  aufgenommen werden.

Dann wird die Berechnung des optimalen Aufnahmezeitpunktes ausgelöst. Fenster b) zeigt nun die Werte wie in Abb.6-2 dargestellt. Der Satellit hat um 23:13:11 seine höchste Elevation von  $78,92^\circ$ . Der Timer zeigt dem Operator an, dass ihm noch 6 Minuten und 50 Sekunden für die Durchführung der Manöver bleiben.



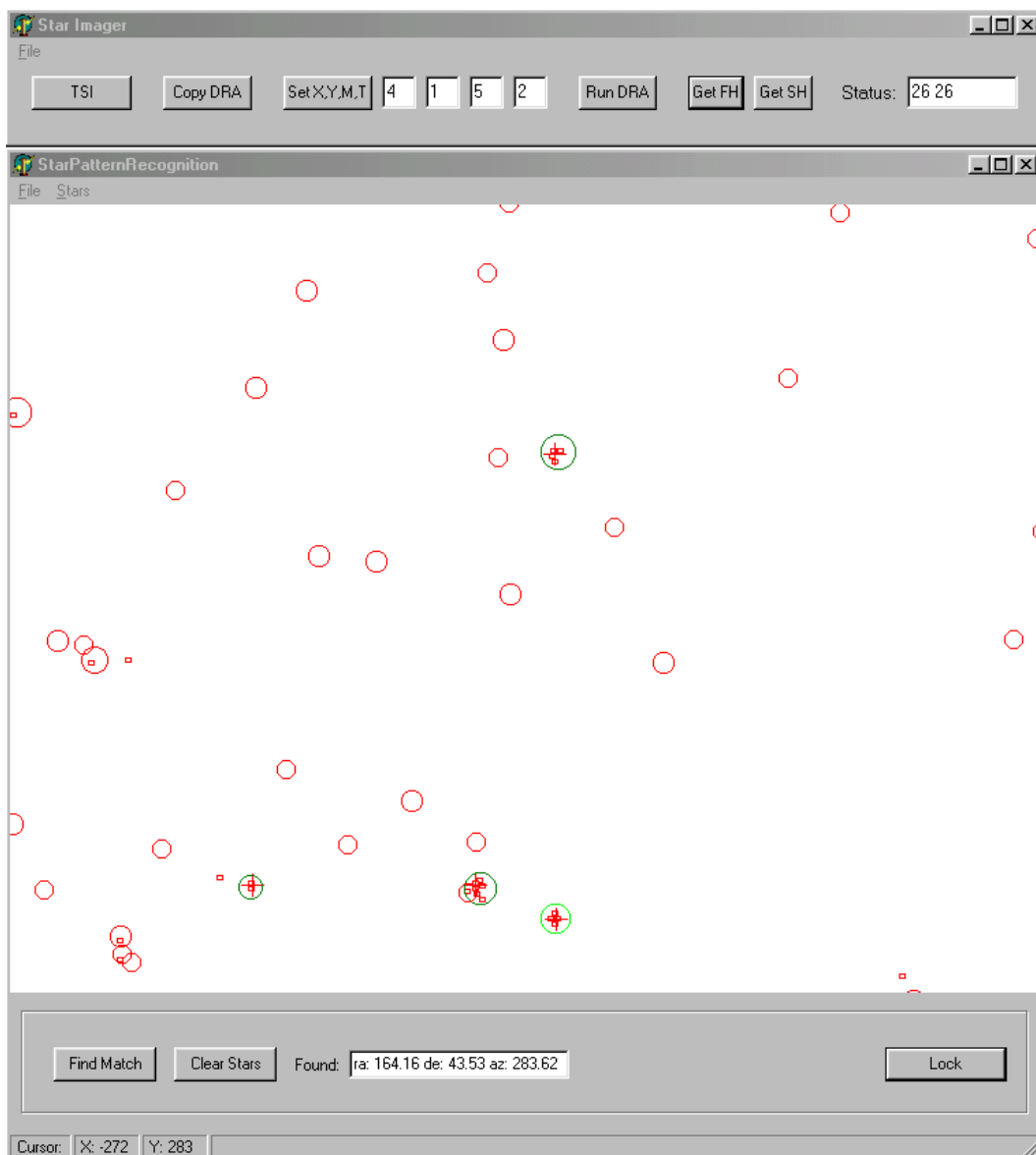
**Abbildung 6-3:** Reduziertes Sternbild in Fenster d)

Beim ersten Funkkontakt werden die Schaltflächen in Fenster c) der Reihe nach betätigt. Der Satellit wird zunächst inertial stabilisiert und ein Sternbild wird aufgenommen. Anschließend wird das Programm zur Bilddatenreduktion vom Bordrechner in die Kamera kopiert. Nach der Festlegung der Parameter für die Datenreduktion wird diese ausgelöst. Zuletzt wird das reduzierte Sternbild zum Boden übertragen.

Da der Satellit sich beim Anflug in Lost-In-Space-Konfiguration befindet, die Lage also beliebig ist, kann es vorkommen, dass sich die Erde, der Mond oder helle Plane-

ten im Blickfeld der Kamera befinden und das Sternbild damit unbrauchbar machen. In diesem Fall kann nach einem Schwenk um die X- oder Y-Achse ein neues Sternbild in Auftrag gegeben werden. Da man für Schwenk und Neuaufnahme etwa ein bis zwei Minuten zusätzlich einkalkulieren muss, stehen in der Regel pro Überflug nur 3 Versuche zur Verfügung um ein verwertbares Sternbild aufzunehmen. Die Einstellung der Parameter für die Bilddatenreduktion muss ebenfalls meist mehrmals angepasst werden. Da die Anpassung aber innerhalb weniger Sekunden erfolgen kann, ist dies weniger problematisch. Durch die Praxis vieler Überflüge haben sich zudem brauchbare Standardwerte für die Parameter Blockbreite (=4), Blockhöhe(=1) und Versatztoleranz (=2) herauskristallisiert, sodass in der Regel nur noch der Schwellwert an die Helligkeit des Sternbilds angepasst werden muss.

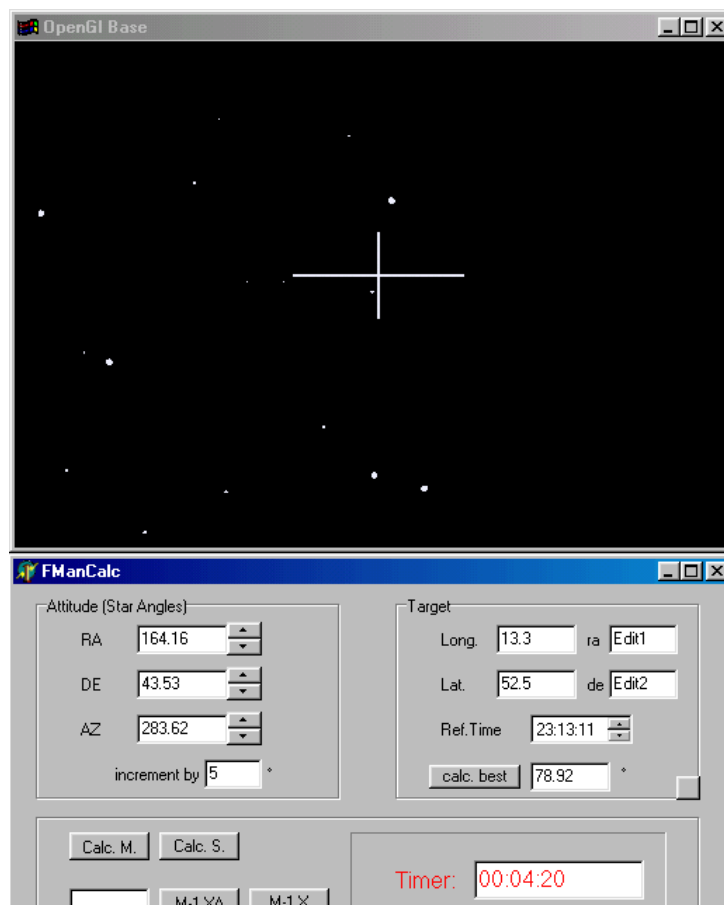
Das reduzierte Sternbild wird nun im Fenster d) angezeigt (siehe Abbildung 6-3). Der Operator muss jetzt im Bild die Sterne markieren, die zur Mustersuche herange-



**Abbildung 6-4:** Ausgewertetes Sternbild in Fenster d)

zogen werden sollen. Sicherheitshalber sollten nur Sterne ausgewählt werden, die aus mehr als einem Block bestehen, da bei ihnen die Wahrscheinlichkeit höher ist, dass es sich um echte Sterne handelt. Da im empfangenen Sternbild nur 4 Sterne dieses Kriterium erfüllen werden auch nur diese markiert.

Anschließend wird die Mustersuche ausgelöst. Da nur 4 Sterne markiert wurden, existieren zunächst mehrere Lösungen, die der Reihe nach angezeigt werden. Der dritte von insgesamt 6 ermittelten Lösungsvorschlägen ist in Abbildung 6-4 dargestellt.

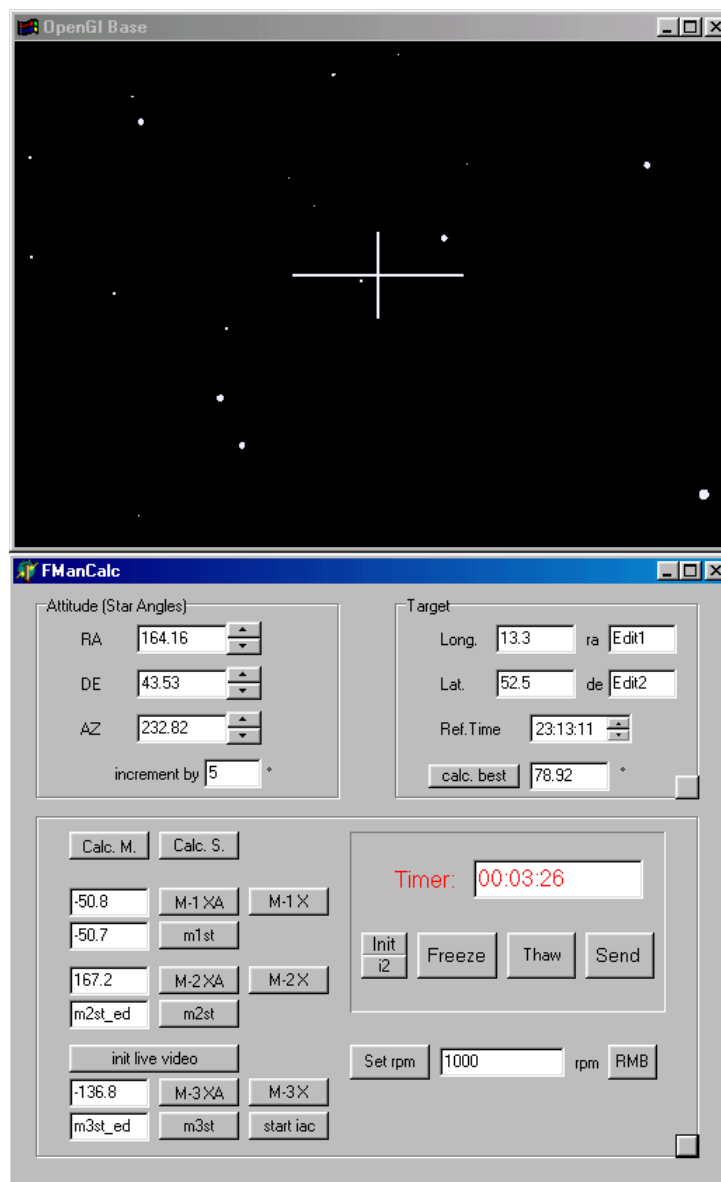


**Abbildung 6-5:** Simulation mit aktueller Satellitenlage

Anhand des Vergleichs von erwarteten und tatsächlichen Sternpositionen stellt der Operator fest, dass es sich hierbei um das passende Sternbild handelt. Durch einen Tastendruck zeigt der Operator dies der Software an. Daraufhin wird die zugehörige Lage als aktuelle Lage von der Simulation übernommen. Die Fenster a) und b) werden aktualisiert und sehen nun aus wie in Abbildung 6-5 gezeigt.

Fenster a) zeigt nun denselben Himmelsausschnitt wie Fenster d). In Fenster b) ist die aktuelle Satellitenlage mit  $RA=164,16^\circ$ ,  $DE=43,53^\circ$  und  $AZ=283,64^\circ$  eingetragen. Der Timer in Fenster b) zeigt an, dass die verbleibende Zeit für die Manöver noch 00:04:20 beträgt. Die Lagebestimmung hat also etwa 2,5 Minuten in Anspruch genommen.

Per Knopfdruck erfolgt nun die Berechnung der Schwenkmanöver, die Ergebnisse der Berechnung werden in den zugehörigen Ausgabefeldern eingetragen. Die Ausführung der Schwenks erfolgt der Reihe nach von oben nach unten. Als erstes wird der Befehl für den Schwenk von  $-50,8^\circ$  um die Bildachse des Satelliten erteilt. Anschließend wird nach einer Wartezeit von etwa 10-20 Sekunden im Abstand von wenigen Sekunden die Telemetrie der kommandierten ACS-Einheit ausgelesen. Dies

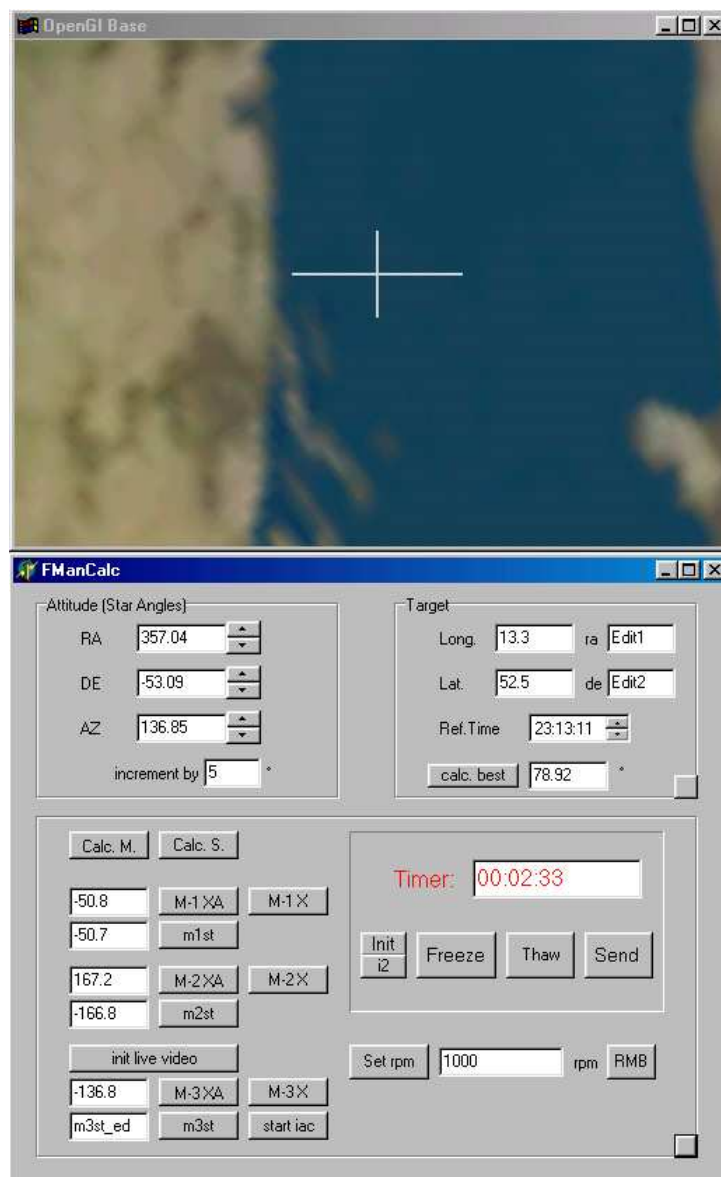


**Abbildung 6-6:** Fenster a) und b) nach 1. Teilmanöver

geschieht solange bis der ausgelesene, also tatsächlich ausgeführte, Winkel dem kommandierten Winkel entspricht. In der Simulation wird zeitgleich derselbe Schwenkwinkel ausgeführt, d.h. das Bild in Fenster a) dreht sich um die Bildachse. Wie bereits an anderer Stelle erwähnt, wurde bei der Animation der Schwenks auf die Modellierung des dynamischen Verhaltens des echten Satelliten verzichtet. Die

Schwenks werden in der Simulation mit einer konstanten Winkelgeschwindigkeit ausgeführt, bieten also keinen Anhaltspunkt für den Operator, wann der Schwenk des echten Satelliten abgeschlossen ist.

Abbildung 6-6 zeigt Fenster a) und b) nachdem das erste Teilmanöver abgeschlossen ist. Das Simulationsbild in Fenster a) ist nun um die  $-50,8^\circ$  gedreht. In Fenster b) erkennt man, dass das letzte Auslesen des Z-Rad/Kreisel-Paares einen ausgeführten



**Abbildung 6-7:** Fenster a) und b) nach 2. Teilmanöver

Winkel von  $-50,7^\circ$  geliefert hat. Die in Fenster b) angezeigte Satellitenlage der Simulation wurde nach dem Schwenk automatisch angepasst.

Da die Lage lediglich mitgekoppelt wurde und nicht neu ermittelt wurde, entspricht die angezeigte Lage nicht exakt der des echten Satelliten. Aufgrund der Drift der faseroptischen Kreisel wächst die Abweichung zwischen simulierter und echter Satellitenlage mit der Zeit stetig an. Aus diesem Grund darf zwischen Lageermittlung und

Abschluss der Manöver nicht zuviel Zeit verstreichen. Eine Messung hierzu ist in Kapitel 8 enthalten.

Analog zum ersten Schwenk wird nun der zweite Schwenk um X-Achse ausgeführt. In Reaktion auf den Befehl bewegt sich das Bild im Fenster a) horizontal. Nach Abschluss des zweiten Schwenks ergibt sich das in Abbildung 6-7 dargestellte Bild. Der Satellit ist nun bereits so ausgerichtet, dass sein Blickpunkt Berlin nach Ablauf des Timers, also in 2 Minuten und 33 Sekunden, überstreicht. Da der Satellit auf der Nachtseite der Erde von Süden nach Norden fliegt, liegt der Erwartungswert für den

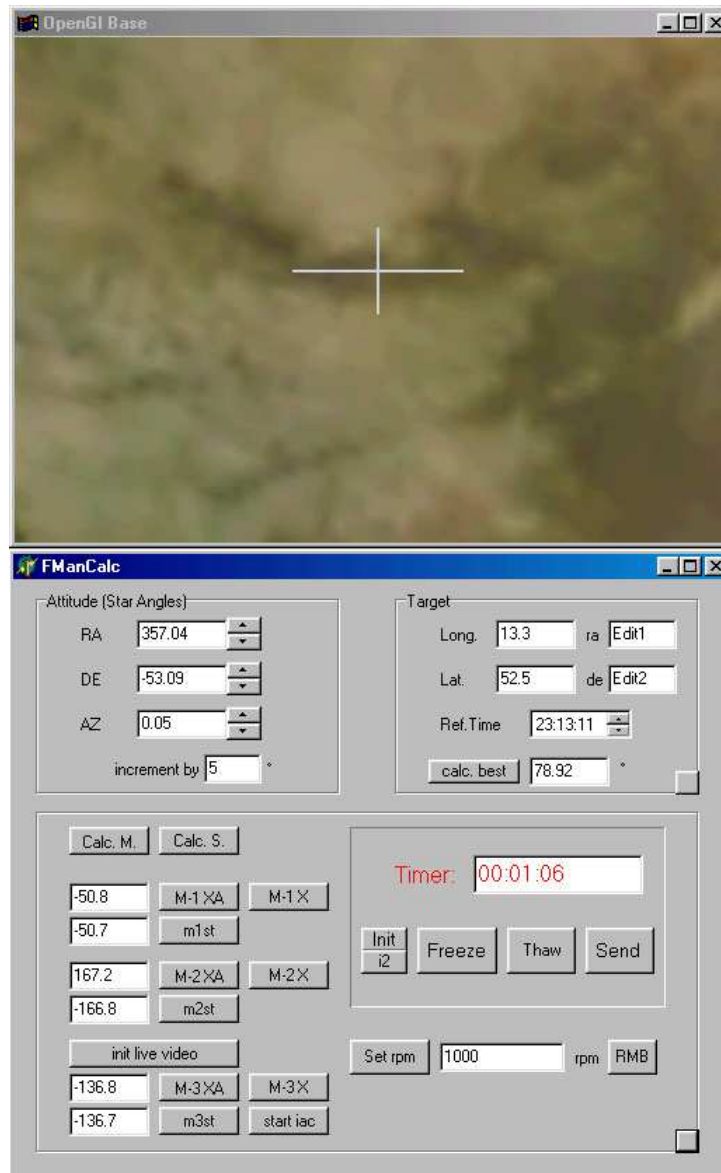


Abbildung 6-8: Fenster a) und b) nach 3. Teilmanöver

Blickpunkt zu dem in Fenster a) dargestellten Zeitpunkt im Mittelmeerraum. Bei Nadir-Ausrichtung des Satelliten wirkt die Erdtextur in der Simulation zwar vergrößert, aber eine Orientierung an Küstenlinien ist dennoch möglich. Der geübte Operator erkennt in Fenster a) die Adria.



Nach Abschluss des zweiten Schwenks wird der Befehl zum initialisieren der Nutzlast erteilt, d.h. die Nutzlastkamera wird auf Live-Video-Modus eingestellt und die S-Band Bildübertragung wird zugeschaltet. Da der Öffnungswinkel der S-Band Antenne deutlich größer ist als der der Optik ( $70^\circ$  gegenüber  $22^\circ$ ), hat der Operator bereits einige Minuten vor dem Erreichen des Ziels das echte Videobild zur Verfügung. Mit der Information aus der Simulation wird es dem Operator erleichtert, die Szene im Live-Videobild richtig zu deuten.

Abbildung 6-8 zeigt die Fenster a) und b) nach Abschluss des dritten Manövers. Das Bild ist damit nun eingenordet, d.h. im Simulationsfenster bewegt sich die Landschaft näherungsweise senkrecht von oben nach unten. Die Vorstabilisierung ist damit abgeschlossen und es bleibt dem Operator noch etwas über eine Minute Zeit, bevor der Blickpunkt Berlin erreicht und die interaktive Lageregelung des Satelliten beginnt.

## 7 Interaktive Steuerungssoftware

### 7.1 Funktionsprinzip

In Kapitel 2 wurden bereits die Probleme beschrieben, die bei der interaktiven Steuerung mit der im Rahmen von [1] entwickelten Software auftreten. Um diese zu umgehen, wurde eine neue Software entwickelt, welche unempfindlicher ist gegenüber kurzen Ausfällen der Funkstrecke und welche eine feinere Dosierung der Drehraten um die Satellitenachsen ermöglicht.

Nach erfolgreichem Abschluss des im vorangegangenen Kapitel beschriebenen Vorstabilisierungsmanövers wird diese Software vom Operator gestartet. Dieser kann daraufhin mit der eigentlichen interaktiven Lageregelung beginnen.

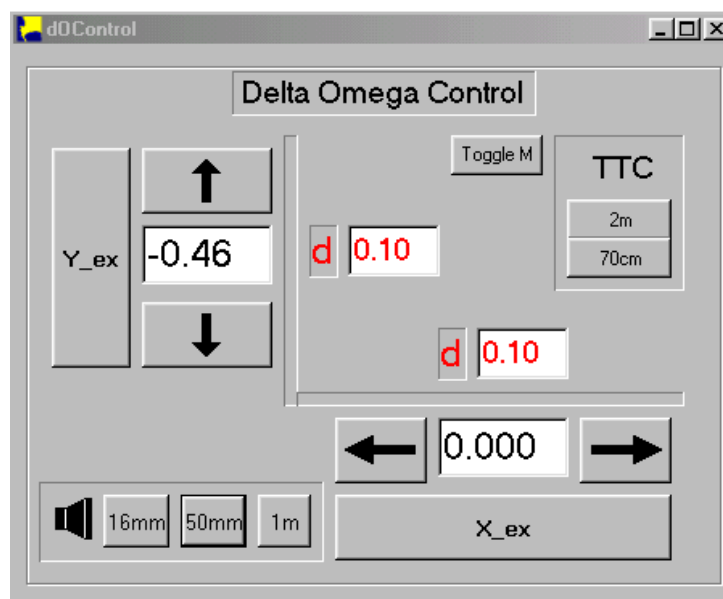


Abbildung 7-1: Interaktive Steuersoftware

Abbildung 7-1 zeigt die Oberfläche des Programms. Die im Fenster angeordneten Schaltflächen können zwar per Mausklick betätigt werden, effektiver ist jedoch das Steuern mittels der Cursor-Tasten auf dem PC-Keybord, da dann der Blick des Operators auf das Live-Bild gerichtet bleiben kann. Gesteuert werden nur die beiden quer zur Bildachse liegenden Achsen X und Y.

Bei jedem Druck auf die Tasten Cursor-Oben/Unten erhöht bzw. verringert sich der Wert für die Y-Drehrate, welcher im großen Eingabefeld links im Fenster angezeigt wird, um einen Schritt. Die Schrittgröße wird dabei im kleineren Eingabefeld daneben in rot angezeigt. Ein Betätigen der Cursor-Tasten bewirkt jedoch noch nicht,

dass ein Befehl zum Satelliten übertragen wird. Dies geschieht erst durch anklicken der Taste „Y\_ex“ oder (bei Keyboardsteuerung) durch Drücken der neben den Cursor-Tasten angeordneten Steuerungstaste („Strg“). Kommt der gesendete Drehratenbefehl aufgrund einer zu schwachen Funkverbindung nicht zum Satelliten durch, so kann der Befehl durch erneutes drücken der Steuerungstaste wiederholt

Objektiv	Schrittweite
16 mm	0,2 %s
50 mm	0,1 %s
1000 mm	0,03 %s

**Tabelle 7-1:** Drehrateninkremente für die 3 Optiken

kann der Befehl durch erneutes drücken der Steuerungstaste wiederholt werden. Damit kommt es nicht zu dem für die Maussteuerung typischen „aufschaukeln“ der Steuerbewegung, die dadurch entsteht, dass der Operator beim Ausbleiben einer Reaktion des Satelliten auf das Steuerkommando die Maus stärker auslenkt als nötig (siehe Kapitel 2).

Die Drehrate um die X-Achse wird analog durch die Tasten Cursor-Links/Rechts verändert. Die Ausführung des Befehls erfolgt wiederum mit der Steuerungstaste. Die Taste ist also keiner der Achsen fest zugeordnet. Es wird jeweils die Achse kommandiert, welche zuletzt verändert wurde. Ein Verändern beider Werte vor dem Absenden des Befehls ist nicht zugelassen.

Im Betrieb hat sich herausgestellt, dass die Reaktionszeit des Satelliten selbst bei idealer Funkverbindung durch die Dauer der Funkübertragung oberhalb von einer Sekunde liegt. Der Operator muss also vor jeder Drehratenänderung geduldig die Reaktion des Satelliten auf das vorangegangene Kommando abwarten. Hier ist die implementierte Keyboardsteuerung ebenfalls von Vorteil, da sie den Operator nicht zu hektischen Manövern verleitet.

Unten links im Fenster sind die Schaltflächen zum Umschalten der Optiken angeordnet. Die Vorstabilisierungssoftware initialisiert die Kamera auf die 50mm-Optik. Die erste Aufgabe während der interaktiven Phase des Nachtüberflugs besteht für den Operator darin, das Ziel im Bild der 50mm-Optik zu zentrieren. Ist dies gelungen, so kann mithilfe der eben erwähnten Schaltflächen auf die 1000mm-Optik umgeschaltet werden und das Zielgebiet genauer abgesucht werden. Für die interaktive Lageregelung mit der 1000mm-Optik sind wesentlich kleinere Veränderungen der Drehraten notwendig. Die Schrittgröße wird daher, wenn der Befehl zum Wechseln des Objektivs erfolgt, automatisch angepasst. In der Praxis haben sich für die verschiedenen Objektive die in Tabelle 7-1 aufgelisteten Schrittgrößen bewährt.

Erreicht der Blickpunkt des vorstabilisierten Satelliten das Gebiet, welches interaktiv abgesucht werden soll, so ist es sinnvoll, zunächst mit einem gezielten Drehraten-

kommando um die Nickachse die schnelle Vorschubbewegung des Satelliten zu kompensieren. Wie schon in Kapitel 2 beschrieben hat der Betrachter des Videobildes nach Einsetzen dieser Gegenrotation den Eindruck, der Satellit würde sich nicht mehr mit einer Geschwindigkeit von 7 km/s über Grund bewegen, sondern wie ein Hubschrauber auf der Stelle stehen. Aus dieser Situation lässt sich die im oberen Abschnitt beschriebene interaktive Lageregelung mit den Cursortasten bequem einleiten. Welche Drehrate zur Kompensation erforderlich ist, hängt von der Position und Bewegung des Satelliten relativ zur Bodenstation ab. Zur Unterstützung des Operators wird die notwendige Drehrate durch die Software anhand von Satellitenposition und -geschwindigkeit automatisch beim Programmstart bestimmt und im Eingabefeld für die Y-Drehrate angezeigt. Der Wert für die Drehrate um X wird hingegen zu Programmbeginn auf 0 gesetzt.

## 7.2 Orbiterfahrung

Die im Abschnitt 7.1 vorgestellte Software wurde vom Autor erfolgreich bei Lageregelungsexperimenten in Stellenbosch, Südafrika, im Rahmen eines mehrwöchigen Forschungsaufenthalts, eingesetzt. Ziel der Experimente vor Ort war es, mittels interaktiver Lageregelung den Satelliten auf Kapstadt auszurichten. Die Versuche wurden in Zusammenarbeit mit der Universität Stellenbosch durchgeführt. Der Empfang und die Aufzeichnung der Bilddaten wurde vom OTB in Overberg übernommen.

Da die Experimente bei Tag durchgeführt wurden, konnten für die Vorstabilisierung wieder die Solarzellenströme ausgewertet werden. Es wurde daher auf eine der beiden in Kapitel 2 vorgestellten Strategien zur Vorstabilisierung zurückgegriffen, nämlich die Ausrichtung der -Z-Seite des Satelliten zur Sonne.

Dieses Verfahren eignet sich sehr gut für den Einsatz auf der Südhalbkugel. Der absteigende Knoten der sonnensynchronen Umlaufbahn des DLR-TUBSAT befindet sich auf der Tagseite der Erde, der Satellit fliegt daher tagsüber immer von Norden nach Süden. Dadurch, dass Kapstadt eine geographische Breite von  $-36^\circ$  hat, ergibt sich hier eine günstigere Konstellation für die interaktive Lageregelung als auf der Nordhalbkugel.

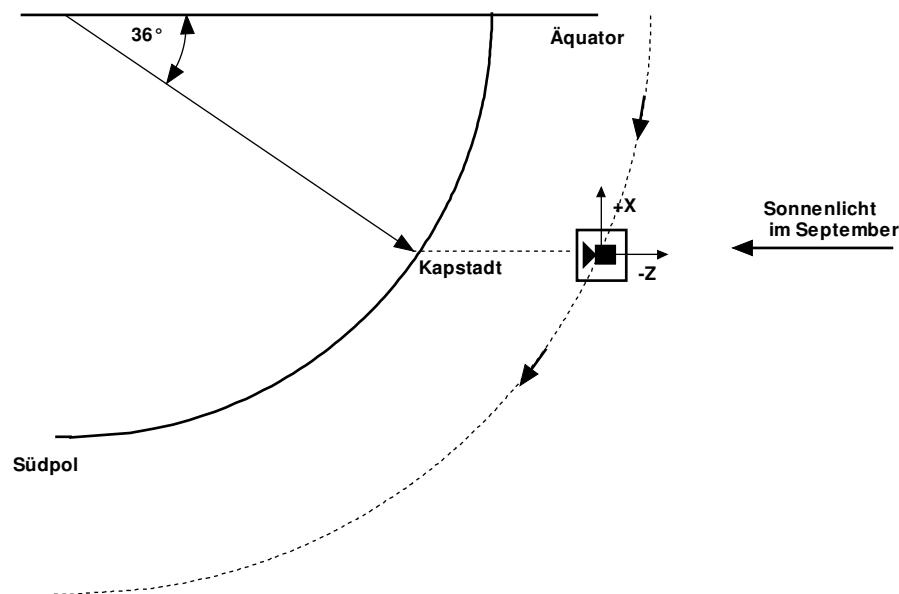
In Abbildung 7-2 ist diese Konstellation abgebildet. Man erkennt, dass der Satellit in Flugrichtung voraus blickt, d.h. dass sein Blickpunkt seinem Fußpunkt vorausseilt. Bei den in Kapitel 2 beschriebenen, in Berlin durchgeführten Experimenten, eilt hingegen der Fußpunkt stets dem Blickpunkt voraus, im Winter sogar so stark, dass dieses Verfahren dann nicht mehr sinnvoll anwendbar ist.

Durch das Vorausseilen des Blickpunkts bleibt dem Operator nun mehr Zeit für die interaktive Suche. Zusätzlich besteht die Chance, dass das Ziel, sofern die Suche zü-

gig abgeschlossen werden kann, zum Zeitpunkt der geringsten Entfernung zum Satelliten aufgenommen werden kann, wodurch eine höhere Bodenauflösung der Aufnahmen erzielt werden kann.

In Abbildung 7-2 ist die Geometrie der Überflüge vereinfachend zweidimensional dargestellt, was suggeriert, dass sich die Sonne in der Bahnebene des Satelliten befindet. Dies war kurz nach dem Start des Satelliten (Mai 1999) näherungsweise der Fall, da sich der absteigende Knoten bei 12 Uhr befand. Bis zum Zeitpunkt der Versuche in Südafrika (September 2001) hat sich die Bahn jedoch gegenüber der Sonne um etwa  $15^\circ$  weiterbewegt, der Zeitpunkt der Äquatorkreuzung des Satelliten hat sich dadurch um eine Stunde in den Nachmittag verschoben. Dies bedeutet, dass zwischen Fusspunkt und Blickpunkt am Ende der Vorstabilisierung ein Ost/West-Versatz von  $15^\circ$  besteht, der korrigiert werden muss, sofern Wert auf eine genaue Nadir-Ausrichtung gelegt wird.

Die Strategie zur Ausrichtung auf Kapstadt ausgehend von dieser vorstabilisierten Lage bestand darin, zunächst bei unkompensiertem Bildvorschub den Moment zu erkennen, wo der Blickpunkt des Satelliten die Küste kreuzt. Dort wurde der Bild-



**Abbildung 7-2:** Vorstabilisierungsstrategie für Kapstadt

vorschub durch Gegenrotation gebremst und durch dosierte Cursor-Steuerbefehle dem Küstenverlauf bis nach Kapstadt gefolgt.

Da die Lage der Bildachse des Satelliten nach der Vorausrichtung auf einige Grad genau bekannt ist können Zeitpunkt und Ort der Küstenkreuzung im Voraus ungefähr bestimmt werden, was dem Operator Timing und Orientierung erleichtert. Für die Berechnung dieser Erwartungswerte wurde eine entsprechende Software entwickelt.



**Abbildung 7-3:** Interaktive Lageregelungsexperimente bei Tag in Südafrika, 1000mm-Objektiv

Aber auch ohne diese Erwartungswerte ist die Orientierung in Echtzeit an der Südafrikanischen Küste einfach, verglichen mit der Situation bei Versuchen im Mittelmeerraum oder der an Ostsee, wie sie in der Vergangenheit mit dem DLR-TUBSAT durchgeführt wurden. So gibt es beispielsweise keine Verwechslungsmöglichkeiten, da es sich um nur eine, klar definierte und relativ gerade verlaufende Küste handelt. Anhand des Küstenverlaufs im eingeordneten Bild lässt sich zudem schnell feststellen, ob man sich westlich oder östlich von Kapstadt befindet. Die Küste westlich verläuft von Nord-West nach Süd-Ost, die Küste im Osten verläuft eher in Ost-West Richtung.

Abbildung 7-3 zeigt beispielhaft zwei der in Südafrika durchgeführten Überflüge. Für die Erstellung der Abbildungen wurden die analogen Videoaufnahmen zunächst digitalisiert. Danach wurden im Abstand von wenigen Sekunden einzelne Frames herauskopiert und per Bildbearbeitungssoftware zusammengesetzt. Die zu den Abbildungen gehörigen Videofilme wurden im Rahmen des 4th Symposium On Small Satellites in Berlin präsentiert.

Bei beiden Überflügen wurde die Küste westlich von Kapstadt gekreuzt und anschließend dem Küstenverlauf Richtung Südosten nach Kapstadt gefolgt. Da die Erprobung der Joystick- bzw. Keyboardsteuerung den Hauptzweck der Versuche darstellte, wurde in beiden Fällen nur kurz in Kapstadt verweilt, um der Küste weiterzufolgen.

Es ist bei beiden Überflügen zu erkennen, dass es mit der neuentwickelten interaktiven Lageregelungssoftware möglich ist, die Lage des Satelliten bzw. die Blickrichtung der Kamera so präzise zu steuern, dass selbst bei Verwendung der hochauflösenden Optik, mit einem Gesichtsfeld von nur  $0,37^\circ \cdot 0,28^\circ$ [4], mühelos dem Küstenverlauf gefolgt werden kann.

## 8 Flugergebnisse

### 8.1 Vorversuche

#### 8.1.1 Driftmessung

Dieser Vorversuch wurde durchgeführt, um zu ermitteln, mit welcher Geschwindigkeit der Satellit im inertial stabilisierten Zustand, also mit einer Winkelvorgabe von  $0^\circ$  für alle drei Achsen, seine Lage verliert. Mit dem neu entwickelten Sternsensor ist es nun erstmals möglich, das Driftverhalten aller drei Achsen im Orbit zu vermessen.

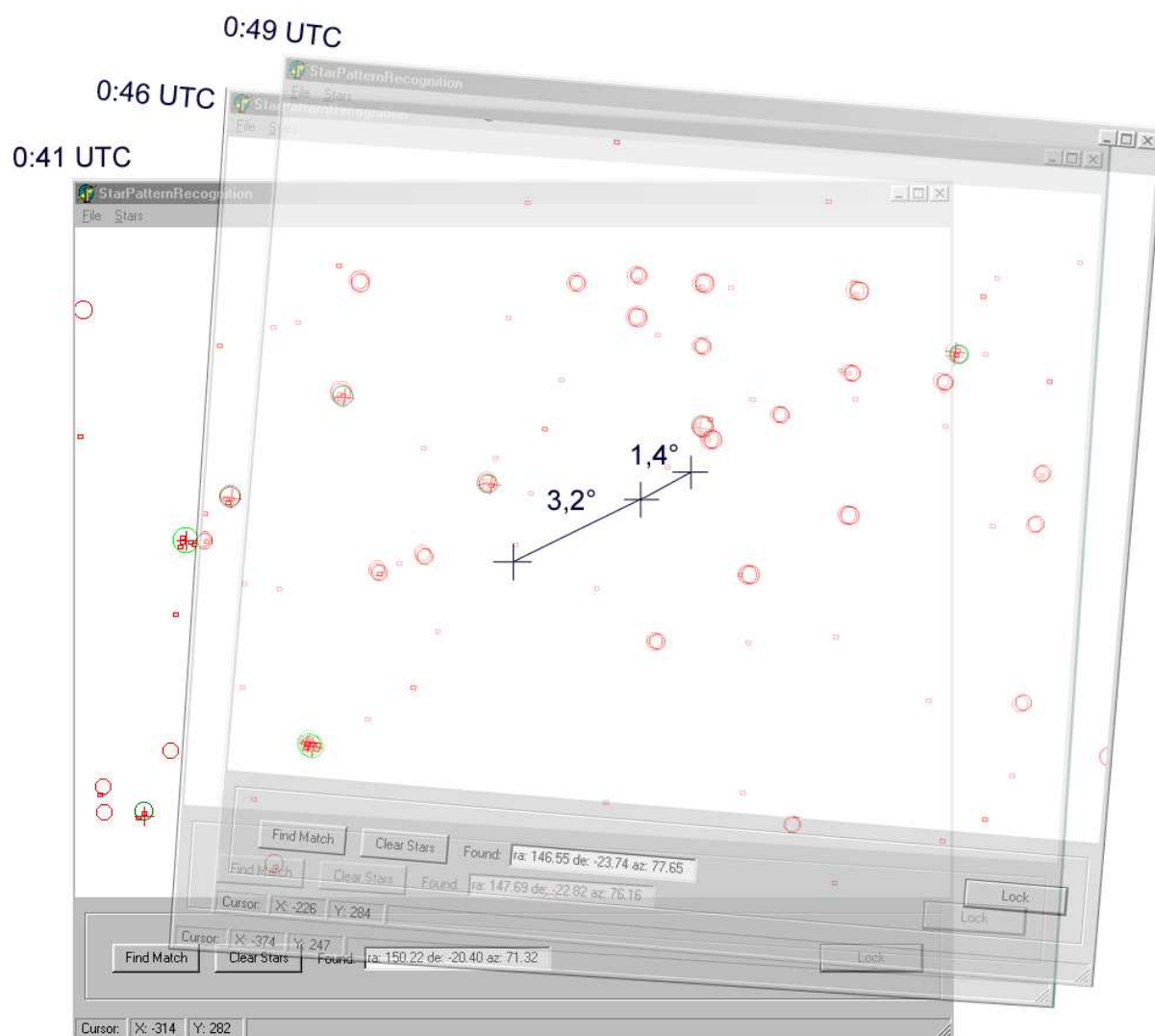


Abbildung 8-1: Driftverhalten des inertial stabilisierten Satelliten



Achse	Fehler nach 5 min [°]	Fehler nach 8 min [°]	Mittl. Driftrate [°/min]
x	2,9	4,1	0,51
y	1,4	2,1	0,26
z	4,8	6,3	0,79

Tabelle 8-1: Driftmessung

Der Driftfehler entsteht durch die Integration des Kreiselrauschens und ist damit veränderlich über der Zeit. Da die Nachtüberflüge in erster Linie für die interaktiven Lageregelungsexperimente genutzt werden sollten, wurden im Rahmen dieser Arbeit nur bei einem Überflug Driftmessungen durchgeführt. Diese wenigen Messergebnisse sollen lediglich einen Anhaltspunkt über die Größenordnung der Driftfehler liefern. Durch die Durchführung einer längeren Messreihe und eine statistische Auswertung derselben ließen sich genauere Aussagen über die maximal auftretenden Driftraten treffen.

Die Betrachtung des Driftfehlers ist vor allem vor dem Hintergrund interessant, dass sich nach Abschluss des Vorstabilisierungsmanövers das ausgesuchte Ziel im Blickfeld der 50mm Optik befinden soll. Damit darf der Abweichungswinkel quer zur Bildvorschubrichtung am Ende höchstens eine Halbe Bildbreite, also  $3.5^\circ$ , betragen. Die Summe aus dem Driftfehler, der Ungenauigkeit beim Setzen der Sternmarkierungen und Abweichungen infolge von vereinfachenden Annahmen bei der Schwenkmanöverberechnung (z.B. die Annahme orthogonaler Satellitenachsen oder die Verwendung des mittleren Erdradius bei der Zielvektorbestimmung) darf diesen Wert folglich nicht überschreiten.

Für den Driftversuch wurde der Satellit zunächst inertial stabilisiert. Unmittelbar nach erfolgter Stabilisierung wurde ein Sternbild ausgelöst. Anschließend wurden im Abstand von 5 bzw. 3 Minuten zwei weitere Sternbilder aufgenommen. Abbildung 8-1 zeigt die drei Aufnahmen, welche mittels einer Bildbearbeitungssoftware übereinander projiziert wurden, um die Driftbewegung zu veranschaulichen.

Tabelle 8-1 zeigt die Driftfehler der einzelnen Achsen. Der Driftfehler um die Z-Achse konnte dabei direkt über die vom Sternsensor gelieferten Azimutwerte bestimmt werden. Die Werte für die X und Y wurden grafisch aus Abbildung 8-1 ermittelt. Der Z-Wert ist mit der größten Unsicherheit behaftet, da die Messgenauigkeit des Sensors um die Bildachse am geringsten ist. Abweichungen beim Markieren der Sterne wirken sich in dieser Achse besonders stark aus. Es wird daher die X-Achse mit einem Wert von etwa  $4^\circ$  nach 8 Minuten als Orientierungspunkt herangezogen. Die Drift liegt damit in der Größenordnung von  $0.5^\circ/\text{min}$ .

Um bei dieser Driftrate die Lageabweichung in einem akzeptablen Rahmen zu halten, muss die Zeit zwischen Aufnahme des Sternbildes und Abschluss der Vorstabilisierung möglichst kurz gehalten werden.

### 8.1.2 Schwenkmanöver von Sternbild zu Sternbild

Um den Gesamtfehler des Vorstabilisierungsverfahrens abschätzen zu können, wurde ein weiterer Vorversuch durchgeführt. Hierbei wurde der Satellit wieder inertial stabilisiert und seine Lage mittels Sternsensor bestimmt. Anschließend wurde als Manöverziel anstelle eines Ziels auf der Erde ein Punkt im Sternbild Cassiopaia, mit den Koordinaten  $RA=15^\circ$  und  $DE=60^\circ$ , in die Manöversoftware eingegeben. Diese Koordinaten wurden gewählt, da sich an dieser Stelle im Sternhimmel mehrere sehr markante Sterne befinden und das Sternbild sich damit besonders leicht auswerten lässt. Außerdem befindet es sich so weit nördlich, dass weder Planeten noch Mond die Aufnahme beeinträchtigen können.

Der Versuch wurde insgesamt dreimal durchgeführt. Oben in Abbildung 8-2 dargestellt ist der Überflug vom 16.12.2002. Links ist das Sternbild der Ausgangslage abgebildet, welches um 22:05 UTC aufgenommen wurde. Die Auswertung durch die

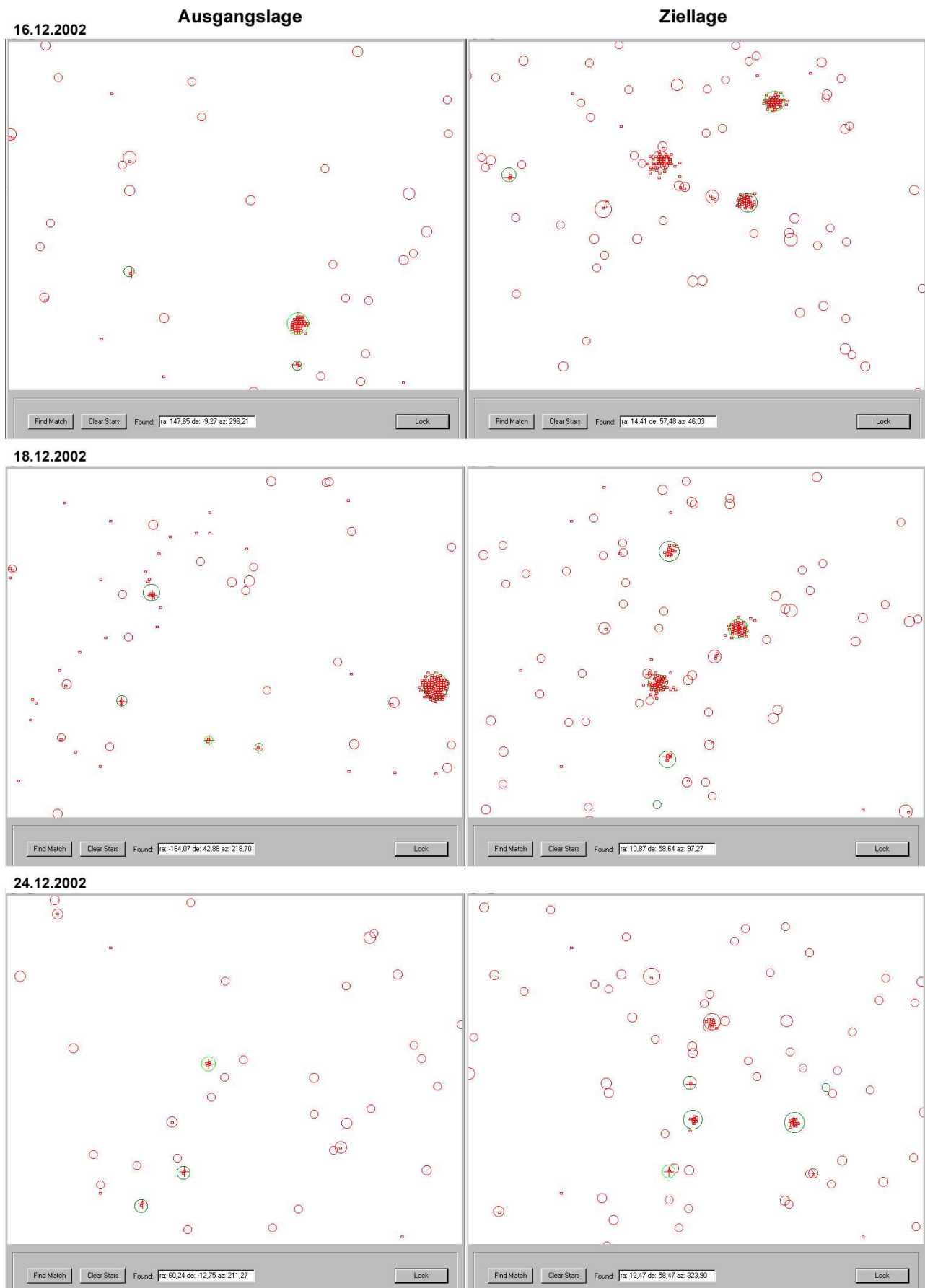
	<i>Ausgangslage</i>				<i>Manöver</i>		<i>Ziellage</i>				
Datum	RA [°]	DE [°]	AZ [°]	t (UTC)	Z [°]	X [°]	RA [°]	DE [°]	AZ [°]	t (UTC)	Abw. [°]
16.12.02	147,65	-9,27	296,21	22:05:34	-1,5	-118,3	14,41	57,48	46,03	22:09:14	2,5
18.12.02	-164,07	42,88	218,70	23:44:02	50,8	-77,1	10,87	58,64	97,27	23:49:13	2,5
24.12.02	60,24	-12,75	221,27	00:32:06	79,8	-81,2	12,47	58,47	323,9	00:35:44	2,0

**Tabelle 8-2:** Schwenkmanöver zum Sternbild Cassiopaia

Sternsensorsoftware ergab eine Lage von  $RA=147,65$ ,  $DE=-9,27$  und  $AZ=296,21$ . Das Ergebnis der anschließenden Manöverberechnung ist in Tabelle 8-2 ablesbar. Es musste zunächst ein Winkel von  $-1,5^\circ$  um die Z-Achse ausgeführt werden und danach ein Winkel von  $-118^\circ$  um X. Nach Abschluss dieser Manöver um 22:09 UTC wurde das Zielsternbild ausgelöst, welches in Abbildung 8-2 oben rechts dargestellt ist. Die Auswertung ergab hier eine Lage von  $RA=14,41$  und  $DE=57,48$ . Die Lage um die Bildachse ist an dieser Stelle nicht von Interesse, da lediglich RA und DE vorgegeben waren. Die Abweichung von der Sollausrichtung wurde anschließend nach Gleichung 4-3 bestimmt. Es ergab sich eine Abweichungswinkel von  $2,5^\circ$  gegenüber dem anvisierten Ziel.

Die beiden weiteren Versuche wurden analog durchgeführt und sind ebenfalls in Abbildung 8-2 dargestellt. Die Ergebnisse aller drei Versuche sind in Tabelle 8-2 zusammengefasst.

Bei keinem der drei Überflüge ergab sich eine Abweichung von mehr als  $2,5^\circ$ . Angesichts der erwarteten hohen Drift des Satelliten ist diese Abweichung erfreulich gering. Die Anzahl der Messungen ist auch hier wieder relativ klein, da nur wenige



**Abbildung 8-2:** Schwenkmanöver zum Sternbild Cassiopeia

Überflüge zur Verfügung standen. Dennoch lässt sich anhand dieses Ergebnisses abschätzen, dass der Fehler des Systems mit hoher Wahrscheinlichkeit innerhalb der im vorigen Unterkapitel angesprochenen Toleranz von  $3,5^\circ$  quer zur Vorschubrichtung liegt.

## **8.2 Interaktive Lageregelung bei Nacht**

### **8.2.1 Vorbemerkung zum Bodensegment**

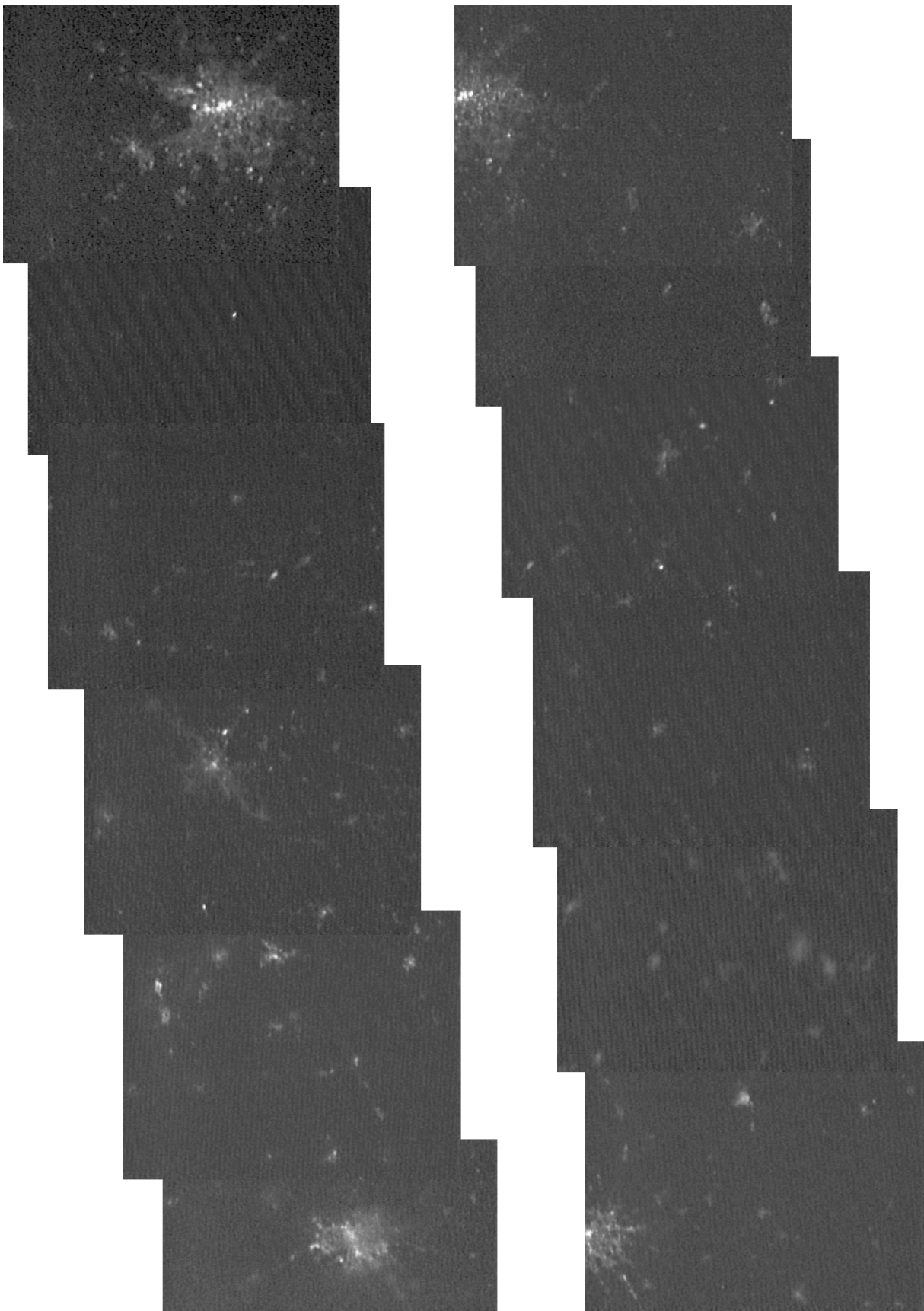
Das Bodensegment für die interaktive Lageregelung bei Nacht wurde gegenüber dem von Schulz [1] beschriebenen Stand geringfügig verändert. Die Übertragung der S-Band Videosignale vom Deutschen Fernerkundungsdatenzentrum Neustrelitz (DFD-Neustrelitz) nach Berlin wurde bereits vor Beginn der Forschungen zu dieser Arbeit von Mitarbeitern des DFD auf ein anderes Verfahren umgestellt. An die Stelle des Real-Servers, einem sog. Video-Streaming-Server, trat dabei ein einfacher Linux-X-Server. Es handelt sich dabei um ein zum Standardumfang eines Linux-Systems gehörendes Werkzeug, was den Export von Fenstern auf entfernte Rechner ermöglicht, sofern diese über ein Netzwerk mit dem Server verbunden sind. Da bei diesem Verfahren unkomprimierte Bilddaten übertragen werden, ist die Anzahl der Frames, welche pro Sekunde übertragen werden können, begrenzt. Bei einer Bildgröße von 200 mal 150 Bildpunkten und 255 Graustufen konnte eine durchschnittliche Übertragungsrate von etwa 5 Bildern pro Sekunde erreicht werden.

Da diese geringe Bildübertragungsqualität den Operator bei der interaktiven Lageregelung einschränkt, wurden im Rahmen dieser Arbeit nochmals verschiedene Video-Streaming-Lösungen untersucht. Bei sämtlichen geprüften Programmen war zwar die Übertragungsrate deutlich höher, zum Teil bis zu 25 Bilder pro Sekunde, jedoch trat bei allen Systemen durch das, für die Bildkompression erforderliche Buffering eine zeitliche Verzögerung von mehreren Sekunden auf. Diese Verzögerung macht die interaktive Steuerung des Satelliten nahezu unmöglich, daher wurde schließlich doch auf das X-Server Verfahren zurückgegriffen.

Parallel zu dem Bildempfang über Neustrelitz wurde auch die 1,20m Parabolantenne auf dem Dach des ILR eingesetzt. Diese Antenne ist für den Empfang des breitbandigen analogen Videosignals eigentlich unterdimensioniert. Da der Satellit bei den Lageregelungsversuchen jedoch direkt auf Berlin ausgerichtet wurde, konnte bei hohen Elevationen ein ausreichend klares Bild empfangen werden.

### **8.2.2 Vorstabilisierung**

Abb. 8-3 zeigt die letzten 40 Sekunden der Vorstabilisierungsphase zweier Überflüge, welche im Februar 2003 im Abstand von wenigen Tagen aufgezeichnet wurden.



**Abbildung 8-3:** Vorstabilisierung bei Nacht auf Berlin, 50mm-Objektiv

Für die Abbildungen wurde das vom DFD-Neustrelitz aufgezeichnete Originalvideo zunächst digitalisiert. Anschließend wurde in Abständen von einigen Sekunden jeweils Blöcke von 15 Frames aus dem digitalisierten Video herauskopiert. Aus diesen Blöcken wurden dann mittels einer astronomischen Bildbearbeitungssoftware Summenbilder errechnet, um den störenden Einfluss von Rauschen und Interferenzen zu minimieren. Die einzelnen Summenbilder wurden zuletzt mittels einer weiteren Bildbearbeitungssoftware zusammengefügt.

Bei beiden Überflügen war Berlin das Ziel der Vorstabilisierung. Durch das in Kapitel 5 beschriebene Verfahren sind die Bilder in Abbildung 8-3 grob eingenordet. Wie bereits erwähnt, kommt der Satellit bei Nacht aus dem Süden, die chronologische Reihenfolge der digitalisierten Frames in der Abbildung ist also von unten nach oben.

Da der Satellit in dieser letzten Flugphase inertial stabilisiert ist, ergibt sich bei jedem Überflug dieselbe Anflugszene, sofern die maximale Elevation der Überflüge nicht zu stark voneinander abweicht. Anhand der Position der bekannten Städte in der Anflugschneise quer zur Flugrichtung kann der Operator bereits lange vor Erreichen des eigentlichen Ziels Lageabweichungen feststellen und gegebenenfalls Korrekturmanöver einleiten. Zunächst wird, unten im Bild erkennbar, die Stadt Prag überflogen. Beim rechts abgebildeten Überflug befindet sich die Stadt zu weit links im Bild. Dies hätte durch ein Schwenk von  $-3,5^\circ$  um die X-Achse des Satelliten korrigiert werden können.

Ungefähr auf halber Strecke zwischen Prag und Berlin wird Dresden überflogen. Da Dresden gegenüber der Achse Prag-Berlin etwas nach Westen versetzt ist, ist die Stadt beim rechten Überflug außerhalb des Blickfelds.

Oben in den Abbildungen ist schließlich Berlin zu erkennen. Beim linken Überflug sogar mit verhältnismäßig geringem Querversatz. Sofort nach Erreichen Berlins wurde bei beiden Überflügen das Kompensationsmanöver eingeleitet.

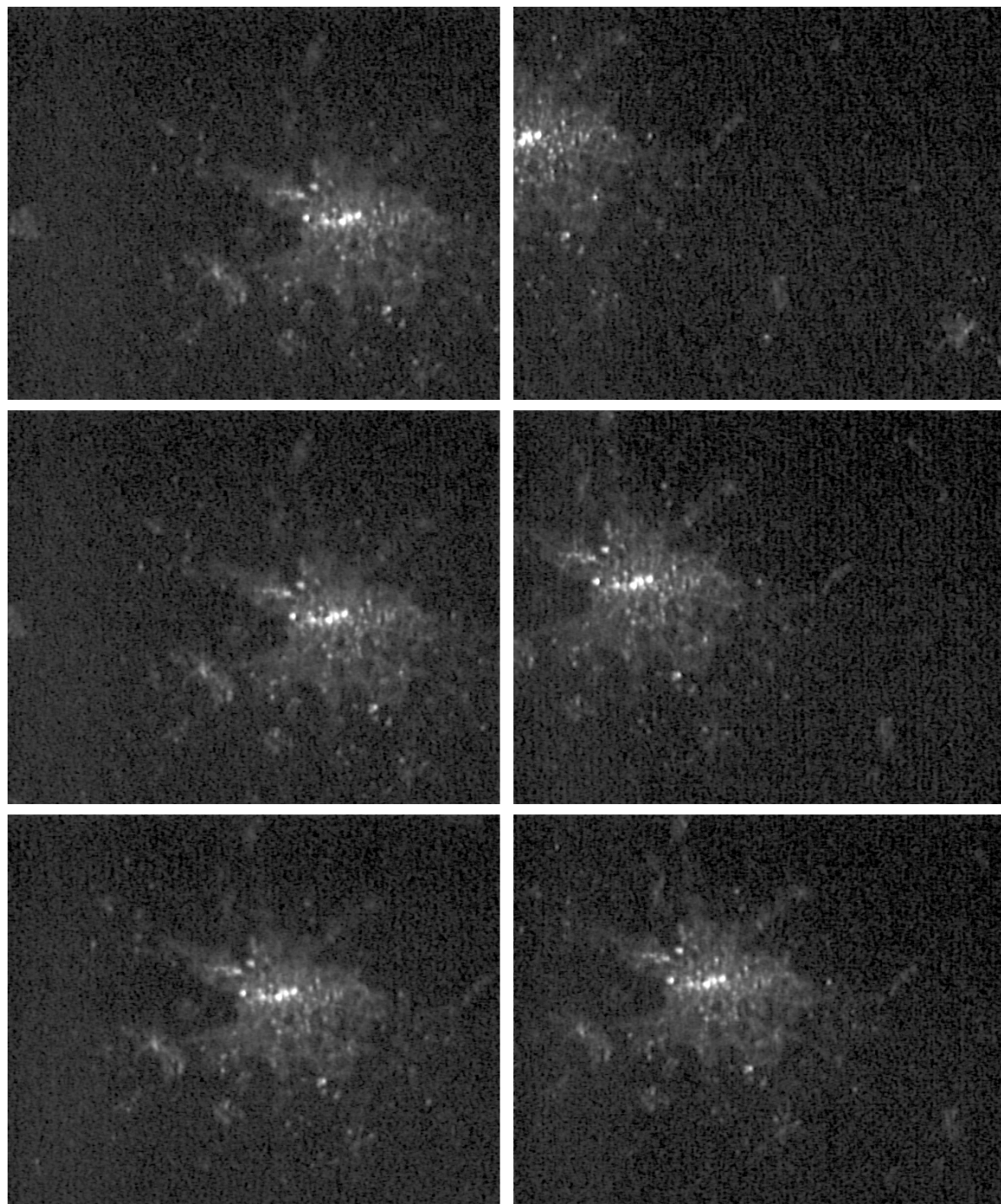
### 8.2.3 Interaktive Lageregelung mit der 50mm Optik

Im Anschluss an das Kompensationsmanöver wurde mittels der in Kapitel 7 vorgestellten Software mit der interaktiven Lageregelung begonnen. Die erste Aufgabe bestand darin, das Ziel, in diesem Fall Berlin, in der Bildmitte des 50mm-Objektivs zu zentrieren. Abbildung 8-4 zeigt diesen Vorgang für beide Überflüge, die zeitliche Reihenfolge der Bilder ist von oben nach unten.

Bei dem links dargestellten Zentriervorgang wurde zunächst die Winkelgeschwindigkeit um die X-Achse des Satelliten, durch zweimaliges Drücken der rechten Cursor-taste und anschließendem Bestätigen durch die Steuerungstaste, um  $-0.2^\circ/\text{s}$  verändert. Unmittelbar danach wurde durch die Tastenfolge „Cursor-Unten“ – „Steu-

rungstaste“ die Winkelgeschwindigkeit in Y um  $-0.1^\circ/\text{s}$  angepasst. Die Kamera bzw. der Satellit reagierte auf die Befehle mit der dargestellten langsamen diagonalen Bewegung Richtung Süd-Osten. In dem Moment, wo die Bildmitte Berlin erreicht hatte, wurde die Bewegung durch Betätigung der jeweils entgegengesetzten Cursortasten und der Steuerungstaste wieder zur Ruhe gebracht.

Bei dem rechts dargestellten Überflug wurde entsprechend vorgegangen, mit dem Unterschied, dass hier eine etwas größere Korrektur in Nord-West-Richtung vorge-



**Abbildung 8-4:** Zentriervorgang mit der 50mm-Optik



nommen werden musste. Nimmt der Zentriervorgang längere Zeit in Anspruch, weil z.B. aufgrund einer schlechten Funkverbindung nicht alle Befehle auf Anrieb durchdringen, so muss die Drehrate um die Y-Achse öfters nachkorrigiert werden. Dies hat den Grund, dass der Blickwinkel auf das Ziel von dem sich entfernenden Satelliten immer flacher wird. Die Gegenrotation muss daher stetig verlangsamt werden um das Ziel im Bild zu halten.

Nach dem erfolgreichen Ausrichten der 50mm Optik auf Berlin wurde schließlich auf das 1000mm-Objektiv umgeschaltet.

#### 8.2.4 Interaktive Lageregelung mit der 1000mm Optik

Bei den Überflügen, welche in den beiden vorangegangenen Unterkapiteln dokumentiert sind, war es nicht möglich, nach dem Umschalten auf die 1000mm-Optik im Bild Lichtquellen zu erkennen, anhand derer die Stabilisierung des Bildes hätte erfolgen können. Vor dem Start des Satelliten wurde die relative Ausrichtung der 50mm- und der 1000mm-Optik vermessen und dokumentiert [4]. Ein Vergleich dieser Messung mit Abbildung 8-4 (unten) ergibt, dass die 1000mm Optik bei beiden Überflügen auf das Berliner Stadtgebiet ausgerichtet war. Zwei der Ursachen dafür, dass dennoch keine Lichtquellen ausgemacht werden konnten, konnten bei diesen Überflügen unmittelbar identifiziert werden: Zum einen ist die 1000mm-Optik mit

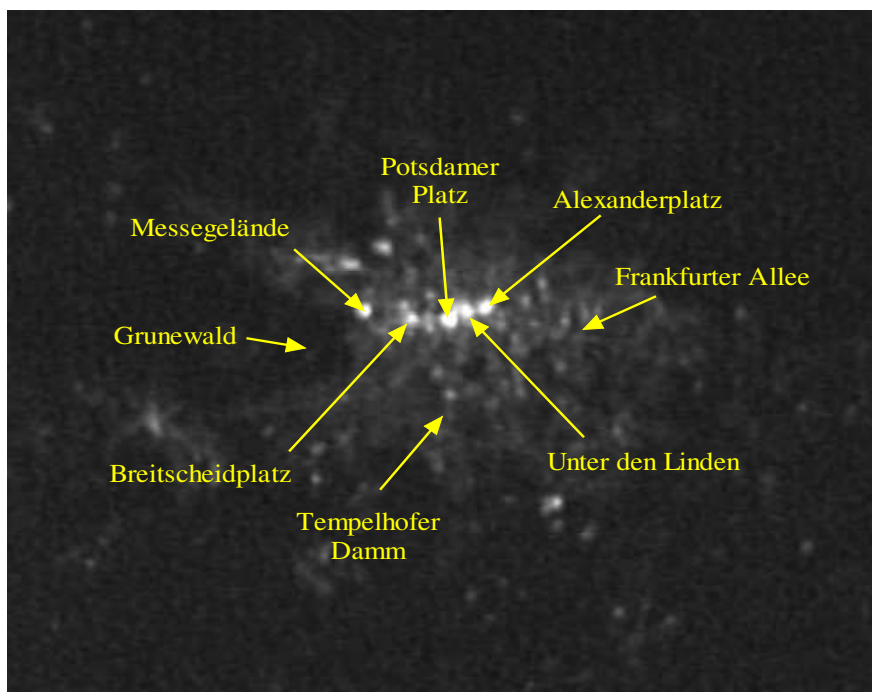


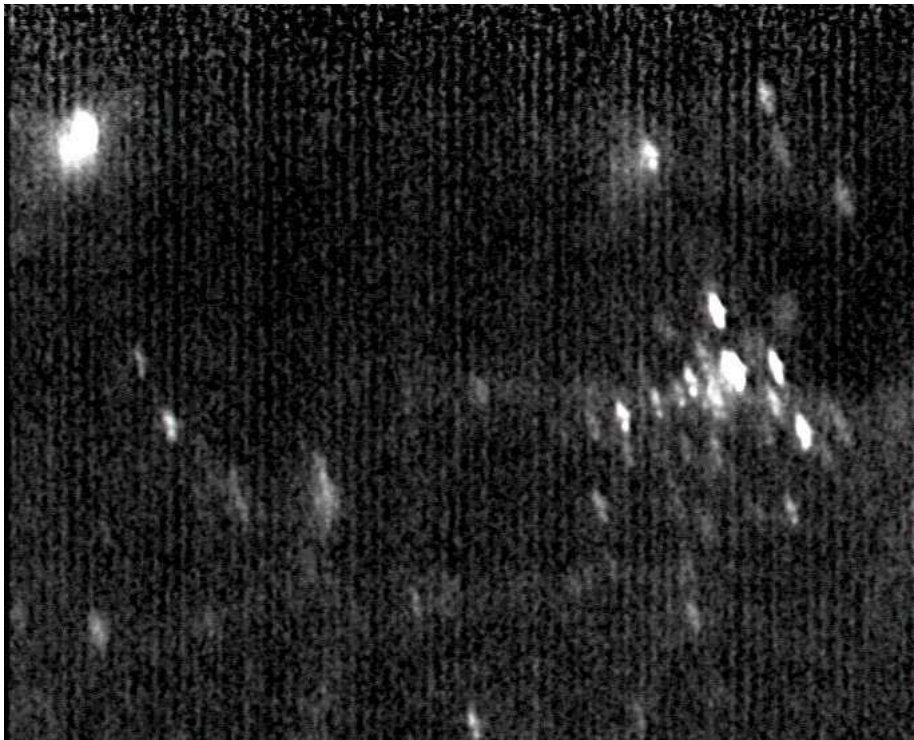
Abbildung 8-5: Berlin bei Nacht (50mm)

ihrem Öffnungsverhältnis von 1/12 und der auf 40ms begrenzten Belichtungszeit (siehe Kapitel 3.2.1) offenbar nicht lichtstark genug für die Aufnahme von Nachtbildern. Zum anderen treten im Live-Videobild starke Interferenzen in Form von dia-



gonalen Streifen hervor, sobald die Kamera aufgrund der schwachen Beleuchtungsverhältnisse die Verstärkung anhebt. Diese Interferenzen werden von anderen, bislang nicht identifizierten, elektronischen Komponenten an Bord des Satelliten verursacht. Bei guten Beleuchtungsverhältnissen am Tage wirken sich die Interferenzen nicht störend auf das Bild aus. Hierin liegt wohl auch der Grund, warum sie bei Tests vor dem Start des Satelliten nicht bemerkt wurden.

Aufgrund der Tatsache, dass diese Überflüge keinerlei verwertbare 1000mm-Bilder lieferten, wurde ein weiterer Versuch unternommen, bei dem nun gezielt die waagrecht verlaufende, aus 5 sehr hellen Lichtquellen bestehende, Kette anvisiert wurde. Abbildung 8-5 zeigt das Ergebnis eines Vergleichs der 50mm-Aufnahmen mit einer Karte von Berlin. Bei der, von links nach rechts betrachtet, ersten Lichtquelle handelt



**Abbildung 8-6:** 1000mm-Aufnahme in Berliner Innenstadt

es sich um den Funkturm oder das Messezentrum, da links darunter der Grunewald als dunkle Fläche deutlich erkennbar ist. Ebenfalls gut erkennbar im Bild ist der Tempelhofer Damm in dessen Verlängerung sich das Brandenburger Tor befindet. Ganz rechts in der Kette befindet sich der Alexanderplatz in Verlängerung der Frankfurter Allee. Bei den beiden restlichen Punkten der Kette handelt es sich vermutlich um den Breitscheidplatz sowie um den Potsdamer Platz.



**Abbildung 8-7:** 50mm-Bild vor dem Umschalten auf die 1000mm-Optik

Für diesen Überflug wurde ein Fadenkreuz auf dem Bildschirm aufgebracht, um den Bereich des 1000mm Blickfeldes innerhalb des 50mm Blickfeldes kenntlich zu machen. Mit diesem Hilfsmittel konnte die in Abbildung 8-6 dargestellte Szene mit dem 1000mm-Objektiv aufgenommen werden. Bei dem Bild handelt es sich wieder um ein Summenbild aus mehreren Frames. Zur Verstärkung der Kontraste wurde es zusätzlich nachbearbeitet, d.h. Tonwerte, Kontrast und Helligkeit wurden angepasst.

Abbildung 8-7 zeigt das 50mm-Bild unmittelbar vor dem Umschalten. Nach dem Umschalten driftete der Satellit noch  $0,3^\circ$  nach Süden bevor er auf der in Abbildung 8-6 festgehaltenen Szene stabilisiert werden konnte. Als rotes Rechteck eingezeichnet ist der Erwartungswert für die Blickrichtung des 1000mm-Objektivs nach dieser kurzen Driftphase (auf der Basis der vor dem Satellitenstart ermittelten Ausrichtung des 1000mm Objektivs [4]). Die tatsächlich mit dem hochauflösenden Objektiv erfasste Szene ist der in Abbildung 8-7 gelb markierte Bereich. Die relative Ausrichtung von 50mm- und 1000mm-Optik weicht damit offenbar geringfügig vom Erwartungswert ab.

Da in dem Live-Bild der 1000mm-Optik neben den beiden hellen Bereichen (Abbildung 8-6, oben links und auf halber Höhe rechts) keine weiteren Objekte zu erkennen waren, wurde auf eine weitere interaktive Erkundung der Stadt verzichtet. Es wurde lediglich diese eine Szene mit vereinzelt kleinen Korrekturen der Drehraten im Bild gehalten.

Die Nachtaufnahme mit der 1000mm-Optik zeigt, dass sich helle, eigentlich punktförmige Lichtquellen als doppelte Dreiecke abbilden. Die doppelte Dreiecksstruktur hat eine Breite von etwa 10 und eine Höhe von etwa 20 Pixeln. Die Lichtstärke des 1000mm-Objektivs reicht offenbar nur aus, um diese Lichtquellen abzubilden. In weniger stark beleuchteten Regionen des Bildes sind hingegen keinerlei Strukturen zu erkennen. Selbst die Reduktion des Rauschens durch das Addieren mehrerer Frames und die Durchführung einer Tonwertkorrektur bringen wenig zusätzliche Bildin-



**Abbildung 8-8:** 1000mm-Aufnahme bei Tag

formation. In Abbildung 8-6, lässt sich gerade noch der Verlauf einiger gut beleuchteter Hauptstrassen erkennen.

Von der hochauflösenden Optik des DLR-TUBSAT war bereits vor Beginn dieser Arbeit bekannt, dass es mit einer dreieckförmigen Point-Spread-Funktion abbildet. Abbildung 8-8 zeigt ein bei Tag aufgenommenes Bild des DLR-TUBSAT mit derselben Optik. Anhand des hellen Objekts im unteren Bereich des Bildes lässt sich die Lichtverteilung einer punktförmigen Lichtquelle gut erkennen. Entlang der Oberkante des Dreiecks befinden sich zwei Intensitätsmaxima, die sich in der Regel gegen die übrigen Pixel durchsetzen. Der Schaden bei einem gleichmäßig ausgeleuchteten Bild ist damit begrenzt auf eine Verdopplung der Konturen in X-Richtung. Ist eine Lichtquelle jedoch so stark, dass diese Maxima in die Sättigung gehen, dann wird das gesamte Dreieck sichtbar.

Bemerkenswert ist, dass dem aufgenommenen Nachtbild nun offenbar eine veränderte Point-Spread-Funktion zugrunde liegt, welche anstelle eines einfachen Dreiecks eine doppelte Dreiecksform liefert. Mögliche Ursache dieser Veränderung des

Abbildungsfehlers könnte die niedrigere Temperatur des Satelliten und damit der Kameraplattform während des Nachtüberflugs sein. Vor dem Erreichen Berlins befindet sich der Satellit bereits etwa 20 Minuten im Erdschatten. Messungen zum Thermalhaushalt des DLR-TUBSAT im Orbit [1] haben gezeigt, dass sich der Satellit während der Schattenphase um etwa 5 °C abkühlt.

### **8.3 Potential der Objektive für Nachtaufnahmen**

#### **8.3.1 16mm-Objektiv**

In der Anfangsphase dieser Arbeit war noch die Verwendung des 16mm Objektivs als Vorfeldsensor vorgesehen. Dies hätte vor allem das Vorstabilisierungsmanöver erleichtert, da sich aufgrund des größeren Öffnungswinkels die zulässige Abweichung nach abgeschlossenem Manöver von 3.5° auf 11° quer zur Flugrichtung erhöht hätte. Außerdem weist das 16mm Objektiv mit 1/0,95 (Öffnungsverhältnis 50mm: 1/1,8) das größere Öffnungsverhältnis auf und ist damit lichtstärker, zumindest

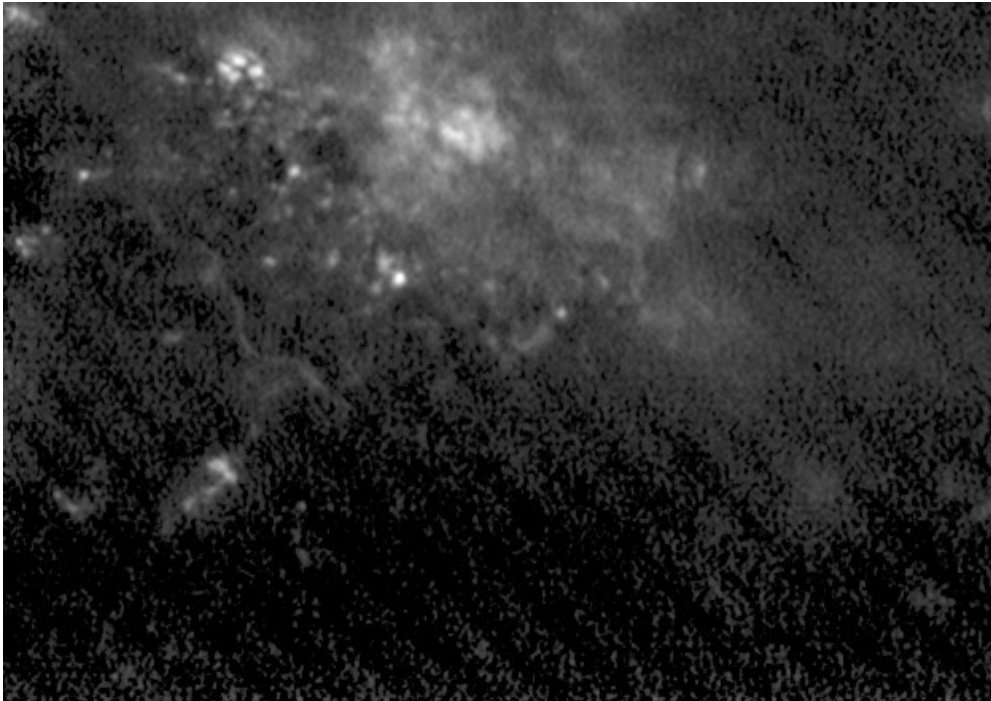


**Abbildung 8-9:** Aufnahme von London (16mm-Objektiv)

dann, wenn es darum geht flächige Motive aufzunehmen, wie z.B. eine vom Vollmond beleuchtete Landschaft.

Es hat sich jedoch gezeigt, dass das Objektiv für Nachtaufnahmen nicht geeignet ist, da abgesehen von großen Städten nur sehr wenig im Bild zu erkennen ist. Bei Abbildung 8-9 handelt es sich um ein 16mm-Bild von London und Umgebung. Abbildung 8-10 zeigt dagegen das 50mm-Bild kurz vor dem Umschalten auf das 16mm Objektiv.

Im 50mm-Bild ist erkennbar, dass das Stadtzentrum und der Nord-Osten der Stadt bewölkt sind, der Süd-Westen jedoch frei ist. Insbesondere gut erkennbar ist der Flughafen Heathrow oben links, sowie ein helles Autobahnstück unten links. Im



**Abbildung 8-10:** Aufnahme von London (50mm-Objektiv)

16mm-Bild hingegen sind nur größerer Städte als helle Wolken zu sehen, die Konturen der Städte sind nicht erkennbar. Auch bei weiteren Aufnahmen von anderen Städten wie Madrid und Paris konnte dieser unscharfe Bildeindruck bestätigt werden. Eine Ursache hierfür ist, dass die Objektive, wie in Kapitel 3 bereits erwähnt, neben ihrer Funktion als Nutzlast- auch als Sternsensoroptiken fungieren sollten. Für die Verwendung als Sternsensor wurden sie absichtlich geringfügig defokussiert. Ein zweiter möglicher Grund für die Unschärfe, vor allem des 1000mm-Objektivs, könnten, wie in Kapitel 8.2 schon angesprochen, Temperatureffekte sein.

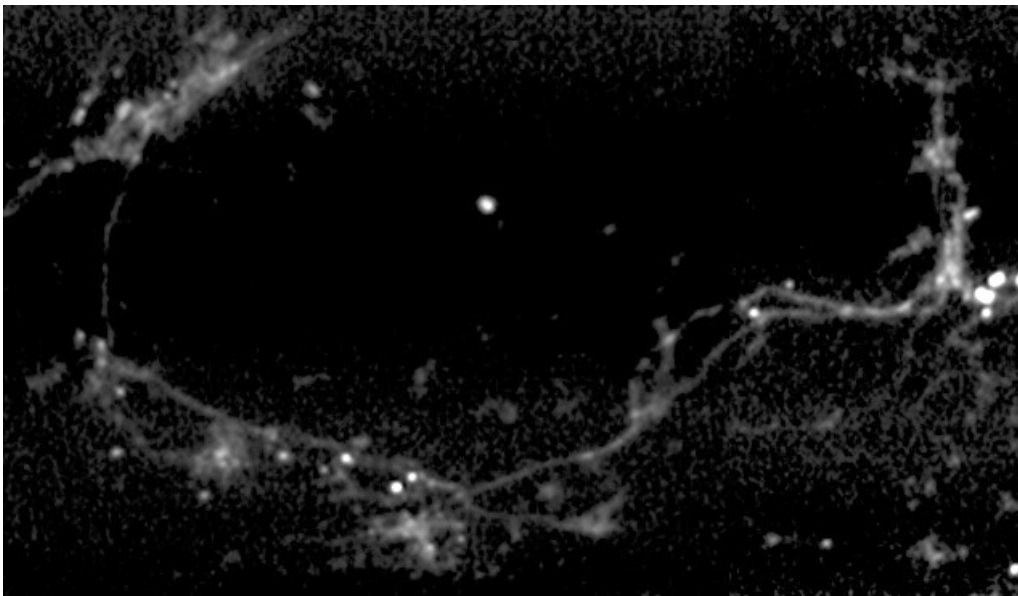
Abbildung 8-9 zeigt außerdem, dass die Lichtstärke des 16mm-Objektivs nicht ausreicht, um abseits der Städte Strukturen der Landschaft sichtbar zu machen. Die Erhöhung des Kontrasts bringt dabei auch keine Verbesserung, es führt lediglich dazu, dass das diagonale Interferenzmuster, wie im Bild erkennbar, noch stärker hervortritt.

### **8.3.2 50mm-Objektiv**

Das 50mm Objektiv lieferte als einziges der drei Objektive unter den, durch Interferenz und Kurzzeitbelichtung, erschwerten Bedingungen ein verwertbares Bild. Im Bild kann man stark beleuchtete Strassen und Plätze sowie andere Objekte wie z.B.

Flughäfen erkennen. Damit ist es möglich während der Vorstabilisierung des Satelliten Städte zu identifizieren.

In Abbildung 8-11 ist nochmals eine mit dem 50mm-Objektiv aufgenommene Szene dargestellt. Es handelt sich um die Südküste Norwegens mit Vororten von Oslo rechts im Bild. Auffällig an dem Bild ist, dass die Strassen sehr viel deutlicher hervortreten als bei anderen Aufnahmen. Eine mögliche Erklärung hierfür ist, dass die



**Abbildung 8-11:** 50mm-Aufnahme der norwegischen Küste

Strassen dieser Gegend zum Aufnahmezeitpunkt schneebedeckt waren und daher das Licht der Straßenlaternen besonders stark reflektierten.

Wie die übrigen Objektive auch, liefert das 50mm-Objektiv ein leicht defokussiertes Bild, die Qualität dieses optischen Kopfs ist jedoch ausreichend, um seine Aufgabe als Vorfeldsensor zu erfüllen.

### **8.3.3 1000mm-Objektiv**

Das 1000mm-Objektiv hat sich unter den gegebenen Bedingungen als für Nachtaufnahmen nicht geeignet erwiesen. Negativ auf die Bildqualität haben sich, wie bei den anderen Objektiven auch, die zu kurze maximale Belichtungszeit von 40ms, die Interferenzmuster und die Unschärfe ausgewirkt. Damit kann das System DLR-TUBSAT nicht wie erhofft hochauflösende Nachtaufnahmen für die in Kapitel 1 angesprochenen Forschungsanwendungen [2][3] liefern. Auch die Nutzung des Systems für Verkehrsüberwachungszwecke ist mit der im Orbit befindlichen Kameraplattform technisch nicht zu realisieren.



## 9 Zusammenfassung und Ausblick

### 9.1 Zusammenfassung

Das erste Ziel dieser Forschungsarbeit war es, den Satelliten DLR-TUBSAT für die interaktive Lagerregelung bei Nacht einsetzbar zu machen und die Funktionsfähigkeit des Systems anhand von Lagerregelungsexperimenten im Orbit zu demonstrieren. Unter interaktiver Lagerregelung versteht man das 1999 an der TU-Berlin erstmals vorgestellte Lagerregelungsverfahren [1], bei dem ein Mensch in der Bodenstation mittels eines Joysticks Steuerbefehle an den Satelliten sendet, während er vor sich auf einem Bildschirm das vom Satelliten übertragene Live-Videobild sehen kann. Er bekommt dadurch in Echtzeit die Reaktion des Satelliten auf die Steuerbefehle vermittelt.

Hinter diesem Ziel steckte die Motivation, einerseits das zeitliche Auflösungsvermögen des Satelliten für bereits in anderen wissenschaftlichen Arbeiten [1] beschriebene Anwendungen (z.B. die Suche nach dem Absturzort eines Flugzeuges oder die Beobachtung von Waldbränden) zu erhöhen, und andererseits neue Anwendungsmöglichkeiten, wie z.B. die Verkehrsbeobachtung, zu erschließen.

Das zweite Ziel dieser Arbeit war die Erforschung der Tauglichkeit der an Bord von DLR-TUBSAT befindlichen Nutzlastoptiken für Nachtaufnahmen. Insbesondere sollte geprüft werden, ob die Bilddaten des Satelliten das Potential besitzen, einen Beitrag zu aktuellen, auf Nachtbildern basierenden Forschungen [2][3] zu leisten. Von besonderem Interesse war hier das 1000mm-Objektiv mit seiner theoretischen Bodenauflösung von 6 m, da bislang keine Nachtbilder anderer Satelliten mit vergleichbarer Auflösung veröffentlicht wurden.

Das Kernproblem, welches zur Erreichung dieser beiden Ziele gelöst werden musste, war das der Vorstabilisierung des Satelliten bei Nacht. Unter Vorstabilisierung versteht man das auf einige Grad genaue Ausrichten des Satelliten auf ein Zielgebiet, welches anschließend interaktiv erkundet werden soll.

Um dieses Problem zu lösen, wurde eine komplexe Vorstabilisierungssoftware entwickelt welche sich aus boden- und bordseitigen Komponenten zusammensetzt. Mit der Software lässt sich der Satellit in drei Schritten von einer beliebigen unbekannten Lage zu Beginn des Überflugs, der sogenannten Lost-In-Space-Konfiguration, in den geforderten vorstabilisierten Zustand überführen. Die drei Schritte sind:

- Aufnahme und Übertragung eines Sternbilds zur Bodenstation
- Ermittlung der Lage des Satelliten durch Auswertung des Sternbilds

- Berechnung und Kommandierung der notwendigen Schwenkmanöver um von der ermittelten zur gewünschten vorstabilisierten Lage zu gelangen

Zu jedem dieser Schritte wurde zunächst eine eigenständige Software entwickelt, anschließend wurden die einzelnen Teile zu der Vorstabilisierungssoftware zusammengesetzt.

Für die Implementierung des ersten Schritts wurde eine sehr kompakte Assemblersoftware entwickelt, die das Sternbild an Bord des Satelliten reduziert, sodass es schnell über die schmalbandige TTC-Funkstrecke des Satelliten zum Boden übertragen werden kann. Der Reduktionsalgorithmus sucht im Bild nach Blöcken heller Pixel, mit vom Operator vorzugebender Größe. Das Ergebnis der Reduktion sind die Koordinaten der gefundenen Blöcke. Diese werden anschließend an die Bodenstation übermittelt und dort von einer weiteren Software wieder in ein Sternbild umgewandelt.

Mit der Suche nach flächigen Objekten anstelle von besonders hellen Pixeln wurde im Rahmen dieser Forschungsarbeit ein Verfahren entwickelt, das auf dem nach über 3 Jahren im Orbit stark strahlengeschädigten CCD-Chip Sterne bis etwa zur Magnitude 5 finden kann. Diese Empfindlichkeit reicht aus, um die Lagebestimmung in beinahe jeder Himmelregion zu ermöglichen.

Für die Ermittlung der Lage des Satelliten wurde eine bodenseitige Sternmustererkennungsoftware entwickelt, welche auf dem Algorithmus des Sternsensors KM1303 basiert. Wichtigster Unterschied zu der Vorlage ist die Tatsache, dass der Operator in den Auswertungsvorgang eingebunden ist. Er muss die Sterne markieren und auch abschließend beurteilen, ob das gefundene Sternbild tatsächlich passt. Im Gegensatz zur Sternerkennung an Bord des Satelliten stehen bei der hier vorgenommenen Implementierung am Boden größere Rechenressourcen zur Verfügung, daher konnte ein wesentlich umfangreicherer Musterkatalog eingesetzt werden. Es wurden im Rahmen dieser Arbeit außerdem eigene Toleranzkriterien für den Mustervergleich entwickelt sowie eine Lageberechnung auf der Grundlage des Dreiecksansatzes eingeführt.

Die für diese Arbeit entwickelte Software zur Berechnung und Kommandierung der Manöver wurde als Echtzeitsimulator unter Einbindung von OpenGL-Grafikroutinen realisiert. Für die Berechnung der Satellitenposition wurde auf die SGP4-Bibliothek von Kelso [13] zurückgegriffen. Die Berechnung der Manöver erfolgt mittels Vektorrechnung. Bei der Kommandierung der Manöver werden diese von der Simulation mitgekoppelt, sodass der Operator den gesamten Vorstabilisierungsvorgang aus der simulierten Satellitenperspektive mitverfolgen kann, was ihm die Interpretation der Szene im echten Videobild erleichtert.

Neben der Lösung des Problems der Vorstabilisierung bei Nacht wurde im Rahmen dieser Dissertation auch die interaktive Steuerung selbst optimiert. Hierfür wurde



wiederum eine neue Software entwickelt. Diese wurde vor dem Einsatz bei Nacht erfolgreich am Tage in Südafrika getestet. Es konnte demonstriert werden, dass es diese Software ermöglicht mit dem 1000mm-Objektiv über mehrere Minuten einer Küstenlinie zu folgen, ohne dieselbe aus dem Blickfeld zu verlieren.

Die Genauigkeit der integrierten Vorstabilisierungssoftware wurde zunächst durch Manöver von einem Sternbild zu einem anderen getestet. Bei keinem der drei durchgeführten Versuchen trat eine Abweichung über  $2,5^\circ$  auf. Für die sich anschließenden interaktiven Lageregelungstests wurde Berlin als Ziel ausgewählt. Die Funktion des Vorstabilisierungssystems konnte durch das Erreichen von Berlin mit dem 50mm-Objektiv bei mehreren aufeinanderfolgenden Überflügen nachgewiesen werden. Auch die Funktion der interaktiven Steuerung konnte durch das erfolgreiche zentrieren des Ziels im Bild nachgewiesen werden. Insgesamt wurde damit bewiesen, dass das System DLR-TUBSAT nun zur interaktiver Lageregelung bei Nacht einsetzbar ist.

Die Auswertung der Nachtaufnahmen, welche im Zuge der Lageregelungsexperimente aufgenommen wurden, hat ergeben, dass das System nicht wie erhofft in der Lage ist, hochauflösende Nachtbilder für wissenschaftliche Anwendungen [2][3] oder für Verkehrsbeobachtungszwecke zu liefern. Es wurden drei Faktoren identifiziert, welche sich negativ auf die Bildqualität aller drei Nutzlastoptiken niederschlagen, besonders aber auf die des 1000mm-Objektivs. Erstens können aufgrund eines Kameradefekts keine Belichtungszeiten oberhalb von 40ms eingestellt werden. Zweitens treten im Nachtbild starke Interferenzen in Form von diagonalen Streifen auf. Und drittens bilden die Objektive, teils aufgrund absichtlicher Defokussierung vor dem Start und teils aufgrund thermischer Effekte im Orbit nur sehr unscharf ab.

## **9.2 Ausblick**

Dass DLR-TUBSAT keine hochauflösenden Nachtbilder liefern kann, liegt an den drei im vorigen Unterkapitel genannten technischen Problemen. Mit der gewonnenen Erfahrung aus diesem Projekt lassen sich diese Probleme für ein Folgeprojekt nun mit überschaubarem technischen Aufwand beheben. Durch Wahl einer Kamera von robusterer Bauart könnten längere Belichtungszeiten ermöglicht werden. Es wäre in diesem Zusammenhang sinnvoll, bereits am Boden die Strahlungsverträglichkeit dieses Systems durch geeignete Versuche zu überprüfen, um das Risiko des Auftretens solcher Defekte zu minimieren. Das Problem der Bildstörungen durch Interferenzen könnte durch Abschirmung der Signalleitungen gelöst werden. Auch hier sollte bereits am Boden unter schwachen Beleuchtungsbedingungen ein End-To-End-Test mit dem vollständig integrierten Satelliten durchgeführt werden, um die Wirksamkeit dieser Maßnahmen zu prüfen. Die Auswirkungen von Temperaturveränderungen des Satelliten auf die Schärfe des hochauflösenden Objektivs könnten schließ-

lich durch einen motorisierten Nachstellmechanismus kompensiert werden, welcher über Funk von der Bodenstation aus betätigt werden kann,.

Ein durch diese Maßnahmen optimierter TUBSAT-Mikrosatellit wäre auf dem Gebiet hochauflösender Nachtbilder ohne Konkurrenz und könnte zu auf Nachtbildern basierenden Forschungen [2][3] entscheidend beitragen. Auch dem Einsatz eines Mikrosatellitensystems zur Verkehrsüberwachung würde man dadurch einen großen Schritt näher kommen.

Bei dem in dieser Arbeit entwickelten Vorstabilisierungsverfahren handelt es sich um eine für den DLR-TUBSAT maßgeschneiderte Lösung. Da das Problem der Vorstabilisierung mit einem bereits im Orbit befindlichen und daher in Bezug auf die Hardware unveränderlichen Satelliten gelöst werden musste, mussten mehrere Umwege gegangen werden, welche bei einer Neuauflage eines solchen Satellitensystems wegfallen würden. Es bietet sich in diesem Fall die Möglichkeit, durch das Mitfliegen eines geeigneten Sternsensors, den Aufwand für die Vorstabilisierung erheblich zu verringern. Es könnte dann zu Beginn des Überflugs sehr schnell an Bord die Lage ermittelt werden und unverzüglich mit den Manövern begonnen werden. Bei der Ausführung der Schwenks könnte dann außerdem auf zwei verschiedene Verfahren zurückgegriffen werden. Einerseits könnten die Schwenks, wie in dieser Arbeit demonstriert, open-loop, d.h. nur unter Verwendung der Winkelregelkreise der Rad-Kreisel-Einheiten ausgeführt werden. Andererseits bietet sich dann auch noch die weitaus elegantere Methode an, den Regelkreis über den Sternsensor zu schließen. Es muss jedoch dafür der Winkel zwischen den Blickrichtungen von Nutzlast und Sensor so gewählt werden, dass der Sensor noch Sterne sehen kann während die Nutzlastoptik auf die Erde ausgerichtet ist. Das Ergebnis wäre dann eine sehr präzise Ausrichtung, sodass ein Ziel auch direkt mit der hochauflösenden Optik anvisiert werden könnte, ohne dass das 50mm Objektiv als Vorfeldsensor benötigt würde.

Bei den Versuchen in Südafrika hat es sich gezeigt, dass sich die interaktive Lagerregelung sehr viel einfacher gestaltet, wenn die Bodenstation, von der aus die interaktive Steuerung des Satelliten vorgenommen wird, auch über eine S-Band-Antenne geeigneter Größe verfügt. Auf diese Weise kann das Videosignal vom Satelliten in guter Qualität direkt vor Ort empfangen und dem Operator verzögerungsfrei angezeigt werden. Für ein Folgeprojekt sollte das Bodensegment in Berlin daher entsprechend ausgebaut werden.

# Symbolverzeichnis

## Abkürzungen

ACS	Attitude Control System
CCD	Charge Coupled Device
DFD	Deutsches Fernerkundungsdatenzentrum
DLR	Deutsches Zentrum für Luft- und Raumfahrt
DMSP	Defense Meteorological Satellite Program
EEPROM	Electrically Erasable Programmable Read Only Memory
EPLD	Electrically Programmable Logic Device
ILR	Institut für Luft- und Raumfahrt
NASA	National Aeronautics and Space Administration
NORAD	North American Aerospace Defense Command
OBDH	On Board Data Handling
OLS	Operational Linescan System
OpenGL	Open Graphics Library
OTB	Overberg Test Range
RAM	Random Access Memory
ROM	Read Only Memory
TTC	Telemetrie- und Telekommando-Einheit
TUBSAT	Technische Universität Berlin Satellit
UTC	Universal Time Coordinated

## Indizes

E	Erdfestes System
G	OpenGL-System
Q	Inertialsystem
S	Satellitenfestes System
X	X-Achse
Y	Y-Achse
Z	Z-Achse

## Konstanten

$k_x$	Winkel pro CCD-Pixel in X-Richtung
$k_y$	Winkel pro CCD-Pixel in Y-Richtung
$T_{\text{Pix}}$	Toleranzwert für Sternmarkierung

**Variablen**

$AZ$	Azimut
$DE$	Deklination
$RA$	Rektaszension
$A$	Rotationsmatrix
$\underline{B}$	Ortsvektor eines Ziels auf der Erdoberfläche
$d_{\text{ccd}_i}$	Abstandswinkel zum Referenzstern (CCD-Chip)
$d_i$	Abstandswinkel zum Referenzstern (Sternenhimmel)
$\underline{q}$	Quaternion
$\underline{r}$	Ortsvektor des Satelliten
$\underline{V}$	Hilfsvektor
$\alpha_{\text{ccd}_i}$	Richtungswinkel (CCD-Chip)
$\alpha_i$	Richtungswinkel (Sternenhimmel)
$\alpha_M$	Schwenkwinkel (Vorstabilisierungsmanöver)
$\beta$	Geografische Breite
$\vartheta$	2. Eulerwinkel
$\theta_g$	Greenwicher Stundenwinkel
$\lambda$	Geografische Länge
$\underline{\rho}$	Zielvektor
$\varphi$	1. Eulerwinkel
$\varphi_{\text{ccd}_{ij}}$	Zwischenwinkel (CCD-Chip)
$\varphi_{ij}$	Zwischenwinkel (Sternenhimmel)
$\psi$	3. Eulerwinkel

## Literaturverzeichnis

- [1] S. Schulz. Interaktive Lageregelung zur Erdbeobachtung mit Mikrosatelliten am Beispiel DLR-TUBSAT. Der Andere Verlag, 2001.
- [2] P. Sutton. A Preliminary Investigation of the Utility of Low-Gain DMSP OLS Nighttime Satellite Imagery for Modeling Population Density.  
<http://www.spatial.maine.edu/ucgis/testproc/sutton/sutton.html>, 2002.
- [3] P. Cinzano. The night sky in the World. [http://www.lightpollution.it/The night sky in the World - Pierantonio Cinzano.htm](http://www.lightpollution.it/The%20night%20sky%20in%20the%20World%20-%20Pierantonio%20Cinzano.htm), 2002.
- [4] M. Steckling. Lageregelung im Bogensekundenbereich am Beispiel des Mikrosatelliten DLR-TUBSAT. Fortschrittsberichte VDI, Reihe 8, Nr. 700, 1998.
- [5] NASA. Earth at Night. <http://antwrp.gsfc.nasa.gov/apod/ap001127.html>, 2003.
- [6] NASA. DMSP-Homepage. <http://dmsp.ngdc.noaa.gov/dmsp.html>, 2003.
- [7] EURIMAGE. LANDSAT Poster, 1996.
- [8] NASA. LANDSAT-Homepage.  
[http://ltpwww.gsfc.nasa.gov/IAS/handbook/handbook\\_toc.html](http://ltpwww.gsfc.nasa.gov/IAS/handbook/handbook_toc.html), 2003.
- [9] KAPPA. Benutzerhandbuch CF 8/1DX und CF 20DX, 1996.
- [10] TU-Berlin. Benutzerhandbuch KM 1301, 1999.
- [11] R. Priebs. Ortung und Navigation. Skript zur integrierten Lehrveranstaltung, 1997.
- [12] TU-Berlin. Projektunterlagen DLR-TUBSAT, 1999.
- [13] T. Kelso. SGP4 Orbitmodell-Quellcode und NORAD-TLE  
<http://www.celestrak.com>, 2003.
- [14] D. Slater. Quaternions. <http://www.resonancepub.com/quaterni.htm>, 1991.

# Lebenslauf

## Persönliche Angaben:

Name:	Jörn-Hendrik Bleif
Geburtsdatum:	22.01.75
Geburtsort:	Berlin
Familienstand:	Ledig
Staatsangehörigkeit:	Deutsch
Eltern:	Dr. Hans-Jürgen Bleif, Diplom-Physiker Hildegard Bleif, geb. Klein, Lehrerin
Geschwister:	Zwei Schwestern

## Schul Ausbildung:

01/1981-11/1981	Mauricewood Primary School, Edinburgh, Schottland
1981-1987	Conrad-Grundschule, Berlin-Zehlendorf
1987-1995	Schadow-Gymnasium, Berlin-Zehlendorf
08/1992-06/1993	Carmel High School, Kalifornien, USA (Schüleraustausch)

## Weitere Ausbildung:

10/1995-11/2000	Studium der Luft- und Raumfahrttechnik an der TU- Berlin
01/1996-11/1996	Grundwehrdienst, Berlin-Gatow
08/1997-10/1997	Praktikum, Siemens AG, Berlin
02/1998-03/1998	Praktikum, Siemens AG, Berlin
07/1998	Abschluss des Vordiploms
07/2000-11/2000	Praktikum, Astrium GmbH, Friedrichshafen
11/2000	Abschluss des Hauptdiploms

## Beruflicher Werdegang:

seit 12/2000	Wissenschaftlicher Mitarbeiter im Fachgebiet Raumfahrtgeräte und -anlagen des ILR der TU-Berlin
--------------	---

## Fremdsprachen:

Englisch, Französisch